Econometric estimation and theoretical modeling of rational stock-market bubbles

Inauguraldissertation

zur Erlangung des akademischen Grades eines

Doktors der Wirtschaftswissenschaften

durch die

Wirtschaftswissenschaftliche Fakultät

der

Westfälischen Wilhelms-Universität Münster

vorgelegt von Diplom-Volkswirt

Benedikt Rotermann

aus Münster

Dekanin: Prof. Dr. Theresia Theurl
Erstgutachter: Prof. Dr. Bernd Wilfling
Zweitgutachter: Prof. Dr. Martin Bohl
Tag der mündlichen Prüfung: 17.11.2014

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Asset prices often exhibit phases of explosive price behavior followed by a crash. In financial economics, a prominent explanation of these stylized empirical facts is the presence of speculative bubbles. These bubbles can occur in different markets like commodity, real-estate and stock markets. Historically, speculative bubbles were considered to be the trigger of substantial economic and financial crises, as shown by the most recent example of the subprime mortgage and financial crisis in 2007-2008. The bursting of the housing bubble in the U.S. and the speculation with risky housing-related securities entailed sharp declines in international financial markets, leading to one of the most severe global recessions since the Great Depression. Therefore, the topic of *speculative bubbles* is still an ongoing and important field of research.

There is a variety of theoretical work trying to explain the emergence of speculative bubbles, their evolution and economic consequences. In addition, there is a wide range of empirical literature providing econometric methods for the identification of bubbles in the data. There is, however, one fact that renders research on speculative bubbles practically difficult. Bubbles are a theoretical construct to explain a price behavior that cannot be explained by standard economic and/or finance the-

ory. In empirical work it is not possible to directly measure and/or observe data of speculative bubbles. Therefore, the basis of empirical work often is a theoretical asset-pricing model that provides a testable relationship that can then be analyzed directly or indirectly by statistical and econometric methods. The existence of several theoretical bubble models and the continuous development of statistical and econometric techniques lead to a further growing empirical literature.

This thesis extends the existing empirical as well as theoretical literature on speculative bubbles. Based on the common theoretical concept of rational behavior, the thesis provides a clear economic model for estimating directly the specification of rational bubbles from the data. The estimation is executed by the use of a sequential Bayesian Monte Carlo methodology that has become increasingly popular in economics in recent years. This approach is then used for further economic and econometric analysis. In doing so, the focus will be on bubbles in stock markets because in the previous bubble literature stock markets are the most intensively studied markets and offer a good database.

The contributions of this thesis are threefold. First, an alternative approach to estimating rational bubbles is presented providing a particular time series of the bubble that can be interpreted unambiguously. Second, the approach is used to obtain further insights into the volatility dynamics of stock prices that are driven by rational bubbles. Third, based on the estimation results, the thesis suggests an alternative econometric specification for rational bubbles that is closely related to economic theory and to financial data.

This thesis is organized as follows. Chapter 2 presents an overview of the existing relevant literature on speculative bubbles and summarizes three theoretical economic concepts trying to explain speculative bubbles in asset prices. Thereby the chapter

focuses on the concept of rational bubbles since it is the basis for the economic model used in this thesis. The second part of this chapter gives an overview of the empirical literature on bubbles and describes concisely the different empirical work trying to find evidence on speculative bubbles in the data.

Chapter 3 provides the econometric procedure in order to estimate periodically collapsing rational bubbles from the data. Based on the present-value model, a non-linear state-space model is presented consisting of the stock price, its fundamental value and a latent bubble component, where the latent bubble component is described by one of the best-known econometric bubble specifications, namely that proposed by Evans (1991). In order to identify the nonlinear state-space model and to estimate the latent Evans-bubble process, so-called particle-filter methods are used. A simulation study shows the reliability of the econometric procedure and the model is applied to data consisting of four major stock-price indices. For all indices, the estimation results indicate the presence of a rational bubble component in the data.

In Chapter 4 the possibility of estimating the latent bubble process is used to analyze the conditional volatility of stock-prices that are driven by Evans bubbles. Therefore, a closed-form volatility formula is derived establishing a link between the bubble component and stock-price volatility. In a simulation study, the estimation procedure presented in Chapter 3 is used to extract the bubble from the data and to compute the specific volatility path. The major finding is that the volatility path is broadly consistent with empirically observed volatility structures during bubbly periods.

Chapter 5 improves some specific features of the general economic framework used in this thesis. A new specification for rational bubbles is proposed that is

empirically more plausible. This new bubble model produces bubbles that burst periodically and deflate stochastically over several periods. The model is applied to artificial and real-world data. Finally, the chapter analyzes theoretical stock-price volatility dynamics under the new bubble specification.

Chapter 6 summarizes the main results of this thesis and offers potential lines of future research. For the implementation of the econometric methods and the empirical applications the software MATLAB is used. The programming codes are compiled in the appendix.

Chapter 2

Literature overview

Although a large part of economic literature deals with the phenomenon of speculative bubbles, there is no consistent definition of the word *bubble* and therefore this expression is widely used to denote different things (see Cochrane, 2005, p. 404). In this thesis we follow a common definition similar to Brunnermeier (2008). We define a speculative bubble as a persistent and explosive increasing divergence between the price of an asset and its fundamental value that is followed by a crash. According to this definition, a temporary explosive price path with a subsequent collapse can be attributed to a speculative bubble.

Regardless of the definition, bubbles are an important field of economic research because they typically indicate temporary and large mispricings of assets. Moreover, bubbles affect the real allocation in the economy through asset prices and a collapse of a bubble impairs the balance sheets of market participants and affects real activity (see Brunnermeier, 2008 and Brunnermeier and Oehmke, 2013). Therefore, speculative bubbles in asset prices are often considered to be a trigger of huge financial and economic crises.

In history, there were several time periods of extreme price behavior that are attributed to bubbles. Garber (1990) refers to the Dutch tulip mania between 1634 and 1637, the Mississippi Bubble between 1719 and 1720, and the South Sea Bubble in 1720 as the first and most famous speculative bubbles in modern literature. Kindleberger and Aliber (2005) extend the list and determine further speculative bubbles, like the late 1920s stock-price bubble, the bubble in real estate and stocks in Thailand, Malaysia, Indonesia and several other Asian countries between 1992 and 1997 or the Dot-com bubble in stock prices in the US between 1995 and 2000. Additional literature on the history of financial crises can be found in Shiller (2000), Reinhart and Rogoff (2009), Brunnermeier and Oehmke (2013) and Hsu (2013), *inter alia*. According to these studies, speculative bubbles and their collapse can be interpreted as a periodic phenomenon.

Following Kindleberger and Aliber (2005), the production on bubble literature is counter-cyclical and as long as economic and financial crises occur, this literature will never lose its relevance. This huge strand of bubble literature can be roughly divided into theoretical and empirical strands of research. The theoretical literature focuses on economic models explaining the emergence and presence of speculative bubbles. The empirical literature deals with evidence on bubbles in the data.

2.1 Theoretical bubble models

There is a considerable amount of theoretical models explaining speculative bubbles in asset prices, see Camerer (1989), Brunnermeier (2008) and Brunnermeier and Oehmke (2013) for an overview. This literature can be divided into three major concepts, based either on rational behavior, irrational behavior or heterogenous beliefs.

In case of heterogenous beliefs, speculative bubbles can occur if agents agree to disagree and if short sales are forbidden. One of the first works concerning heterogenous-beliefs bubbles is the paper of Harrison and Kreps (1978). They present a model in which the investor's disagreement about the fundamental value of a stock can lead to such a speculative phenomenon. Investors are willing to pay a price which contains a bubble component because they expect to sell it later to a more optimistic investor at a higher price and realize capital gains. Heterogenous beliefs can be caused by many reasons. For example, in the model of Scheinkman and Xiong (2003), they arise from overconfident investors who differ in the interpretation of public signals. An overview of further literature can be found in Xiong (2013).

In the second class of models speculative bubbles can occur because of irrational behavior of some market participants. For example, the unpredictability of the behavior of irrational noise traders creates a risk in asset prices that deters rational arbitrageurs with short horizon from trading against an emerging bubble (see De Long et al., 1990 or Shleifer and Vishny, 1997). In the model of Abreu and Brunnermeier (2002) and Abreu and Brunnermeier (2003), a bubble emerges and persists over a substantial period due to a synchronization problem among rational arbitrageurs. Exuberant irrational traders believe in a permanently growing economy and the stock price exceeds its fundamental value and a bubble occurs. In this situation rational arbitrageurs become sequentially aware of this bubble but they do not know whether this information is available to other arbitrageurs. Owing to this synchronization problem the rational arbitrageurs are not able to immediately attack the bubble and in the model equilibrium they ride the bubble for some time after they become aware of the mispricing (see Abreu and Brunnermeier, 2003).

Although speculative bubbles in asset prices can be explained by irrational behavior and heterogenous beliefs, there is recent literature attributing bubbles only to rational behavior. In this general model setup all investors are homogeneous with rational expectations and bubbles can be explained in the context of the standard present-value model. Since the concept of rational behavior is crucial for this thesis, we describe the model of strictly rational bubbles in more detail.

2.2 Present-value model and rational bubbles

In the linear present-value model with rational expectations the price of a stock at date t, P_t, is given by the Euler equation

$$P_t = \frac{1}{1+r}\left[E_t(P_{t+1}) + E_t(D_{t+1})\right], \tag{2.1}$$

where D_{t+1} is the stock dividend payment between t and $t+1$. $E_t(\cdot)$ denotes the conditional expectation operator based on all information available to market participant as of the date t. r is the *required rate of return* that is just sufficient to compensate investors for the inherent riskiness of the stock (see Campbell et al., 1997; Cuthbertson and Nitzsche, 2004). To solve the expectational difference equation (2.1) we substitute future prices forward repeatedly and obtain

$$P_t = \sum_{i=1}^{\infty}\left(\frac{1}{1+r}\right)^i \cdot E_t(D_{t+i}) + \lim_{n\to\infty}\left(\frac{1}{1+r}\right)^n \cdot E_t(P_{t+n}). \tag{2.2}$$

Ruling out speculative bubbles by assuming validity of the transversality condition

$$\lim_{n\to\infty}\left(\frac{1}{1+r}\right)^n \cdot E_t(P_{t+n}) = 0, \tag{2.3}$$

we obtain the unique fundamental stock price

$$P_t = P_t^f = \sum_{i=1}^{\infty} \left(\frac{1}{1+r} \right)^i \cdot E_t(D_{t+i}). \tag{2.4}$$

The basic idea behind a rational bubble is that there are mathematical expressions B_t that are (1) consistent with the limit-term appearing on the right-hand side of Eq. (2.2), and (2) that the stock-price process in Eq. (2.2) satisfies the Euler Eq. (2.1):

$$P_t = P_t^f + B_t = \sum_{i=1}^{\infty} \left(\frac{1}{1+r} \right)^i \cdot E_t(D_{t+i}) + B_t. \tag{2.5}$$

In view of Eq. (2.5), we interpret the rational bubble B_t as the deviation of the current stock price P_t from its current fundamental value P_t^f. The entire class of solutions to the Euler Eq. (2.1) is given by Eq. (2.5) in which B_t is any random variable satisfying the (discounted) martingale property

$$E_t(B_{t+1}) = (1+r) \cdot B_t \qquad \text{or, equivalently,} \qquad B_t = \frac{1}{1+r} \cdot E_t(B_{t+1}). \tag{2.6}$$

It becomes obvious that the bubble B_t contains a self-fulfilling character because the bubble is defined by its own expectations. Because of the (discounted) martingale property the bubble implies rationality. A rational investor is willing to buy a stock which contains a bubble, because the bubble incurs an interest at the required rate of return r (see Cuthbertson and Nitzsche (2004), p. 400).

The condition in Eq. (2.6) implicates some restrictions regarding the existence of rational bubbles. First, negative bubbles can be excluded within this framework because negative bubbles would lead to negative prices at some indefinite future date. Rational bubbles imply explosive conditional expectations and an investor cannot rationally expect a stock price to become negative (see Diba and Grossman, 1988b).

Therefore, by the argument of backward induction the bubble cannot exist at any point in time. Similarly, a bubble can only occur if each investor's horizon is shorter than the time period when the bubble is expected to burst (see Cuthbertson and Nitzsche, 2004, p. 401). If investors expect the bursting of the bubble at a certain point, by backward induction, the bubble must burst immediately. However, if the bubble does not exist from the beginning (that is $B_0 = 0$), it follows from Eq. (2.6) that it is not possible that the bubble starts within the model. Consequently, an infinitesimal small positive bubble must already be part of the stock price. Finally, positive bubbles can only occur in assets where no upper limit on prices exists. For instance, the presence of substitutes limits the price of a certain commodity. At some point in time the bubble would strongly increase the price of the commodity so that it would be substituted by alternative goods so that the bubble vanishes (see Brunnermeier, 2008).

A further theoretical argument that rules out rational bubbles can be found in Tirole (1982). Using a rational expectation equilibrium argument he shows that bubbles do not exist in an economy with a finite number of agents. In such an economy, no trader is willing to buy an asset that incorporates a bubble because he does not expect any gain from it (see Tirole, 1982). This argument does not hold in an overlapping generations model. Tirole (1985) shows that bubbles can exist in this framework as long as the required rate of return is lower than or equal to the growth rate of the economy.

Despite these restrictions, the rational bubble model can explain two important stylized empirical facts of speculative bubbles. First, the model may explain an explosive increasing price path since the bubble component B_t in Eq. (2.6) has to grow in expectation at a rate $(1 + r) > 1$. Second, as long as the bubble component

satisfies the martingale property a collapse of the bubble is not excluded by this model. However, to be consistent with theory, this collapse must be unpredictable for rational market participants. This can be achieved by a stochastic modeling of the bubble component B_t in Eq. (2.6), where the bubble is exogenous to fundamentals and depends only on the bubble component at date $t-1$. The best-known rational bubble models are the bursting bubble in Blanchard and Watson (1982) and the periodically collapsing bubble in Evans (1991). Both bubble specifications are consistent with the martingale property and produce an explosive price path but with the possibility of an unpredictable collapse.[1] As a result, Eq. (2.5) and Eq. (2.6) highlight a particularly useful characteristic of the concept of rational behavior. This model provides a well-defined framework in which the bubble represents a specific mathematical solution that has to satisfy specific mathematical properties. Therefore, the model is a good theoretical basis for empirical application since it offers various testable relationships for identifying speculative bubbles in the data.

In this thesis the focus will be on the exogenous bubble type. However, to complete the literature overview on rational bubbles, we will also briefly look at an alternative type of rational bubbles. Froot and Obstfeld (1991) proposed a specific type of rational bubbles that, in contrast to exogenously defined bubbles, depends exclusively on fundamentals. These so-called intrinsic bubbles are defined as a strictly positive, nonlinear function of dividend payments satisfying the (discounted) martingale property and hence the Euler equation. This bubble is called intrinsic because its dynamics are caused entirely by the fundamentals (see Froot and Obstfeld, 1991). Ikeda and Shibata (1992) and Ikeda and Shibata (1995) suggest a similar type of fundamentals-dependent bubbles. Their rational bubble is defined

[1] We will describe these models in more detail in Chapter 3 and 5.

in a continuous-time framework in which the stock price is specified as a function of dividend payments as well as of time. In an extension, Ikeda and Shibata (1992) also incorporate crash risks to allow for a partially crashing bubble. Within this framework, the authors analyze the influence of the fundamentals on the bubble dynamics and the volatility of stock prices.

2.3 Empirical evidence on bubbles

Based on the theoretical literature on speculative bubbles there exists a continuously growing number of empirical tests and experiments to find evidence on speculative bubbles in the data and to confirm theoretical models on bubbles. Here, we will only focus on the most important works in the field.

We start with empirical literature trying to confirm theoretical models as evidence on speculative bubbles. The survey of Vissing-Jorgensen (2004) uses historical data, statistical tests and regressions to find evidence on investor beliefs and actions and determines whether assumptions made in behavioral asset-pricing models are valid. The authors find empirical support on the assumption that noise trader risks contribute to limit arbitrage supporting the explanations on bubbles by De Long et al. (1990) or Shleifer and Vishny (1997). Similar results are found in Baker and Wurgler (2007). Beside this survey, there also exists some experimental evidence on bubbles. Smith et al. (1988) study the trading of risky assets whereby all investors know the distribution of the dividends that are paid at the end of each period. In the majority of their experimental settings they observe the phenomenon of bubbles that can be interpreted in the context of Tirole (1982), that is bubbles emerge because each trader believes to be able to sell the asset and to realize profits in the final periods. Brunnermeier and Morgan (2010) present a model of clock games to

analyze experimentally the individual's trade-off between gains from waiting versus the risk of being preempted. Their results are consistent with the predictions of the irrational bubble model proposed by Abreu and Brunnermeier (2003) (see Brunnermeier and Morgan, 2010).

In addition to the above-mentioned studies that find empirical and experimental evidence in favor of certain theoretical bubble models or at least in favor of certain behavioral assumptions within these frameworks, a second strand of empirical work exists on bubble tests trying to identify rational bubbles in the data.[2] As mentioned above, the rational bubble model offers some testable relationships that can then be analyzed directly or indirectly by various statistical and econometric methods.

A crucial problem when testing for rational bubbles is that the asset's fundamental value is unknown and that the bubble component is not observable. In general, we can distinguish between (a) tests that explicitly use the theoretical properties and the structure of rational bubbles for their detection and (b) tests that attribute the rejection of the present-value model to rational bubbles. A famous example of the latter type of tests are the variance-bound tests proposed by Shiller (1981) and LeRoy and Porter (1981). Based on Eq. (2.4), Shiller argues that the observed volatility of a stock price has to be less or equal to the volatility of a perfect forecast of prices obtained by the standard present-value model. An application of the variance-bound test to the data rejects the inequality so that stock-price volatility is higher than attributed to fundamentals. Similar testing procedures are described in Campbell and Shiller (1987), Campbell and Shiller (1988) and West (1988). All these tests find evidence in favor of excess stock-price volatility without making bubbles responsible for these results. However, although these tests were not developed as

[2] An overview of some classical tests for rational bubbles can be found in Gürkaynak (2008) and Flood and Hodrick (1990).

bubble tests, there is literature attributing excess volatility to the presence of speculative bubbles (see Blanchard and Watson, 1982; Tirole, 1985; Flood and Hodrick, 1986). Moreover, the articles by Campbell and Shiller (1987) and Campbell and Shiller (1988) are important for further empirical work on rational bubbles. Based on a log-linear approximation of the standard present-value model these authors use a vector autoregressive methodology and thus provide a procedure for estimating the fundamental value of a stock. This approach is part of a variety of more recent empirical works (see, among others, Wu, 1997; Al-Anaswah and Wilfling, 2011; Phillips et al., 2011; Brooks and Katsaris, 2005).

For a bubble test to be useful it is important that the bubble is at least in the set of the alternative hypothesis when the test rejects the standard model (see Gürkaynak, 2008). A testing procedure in this sense is the two-step bubble test by West (1987). The test consists of direct and indirect parameter estimates needed to calculate the closed-form solution of the Euler equation. The key idea is that under the null hypothesis of *no bubble*, both estimates should not differ significantly from each other. Using data of the S&P 500 and the Dow Jones, the author rejects the null hypothesis indicating the presence of speculative bubbles.

The empirical literature using the theoretical properties of rational bubbles is more extensive. One of the first and best known procedure stems from Diba and Grossman (1988a). Based on the present-value model, the authors derive the following results. If the dividend generating process is nonstationary in levels, but first differences of dividends and other unobservable fundamentals are stationary, then stock prices are nonstationary in levels, but stationary in first differences. If the bubble is described by a simple nonstationary first-order autoregressive process satisfying Eq. (2.6), this relationship no longer holds and differencing stock prices

a finite number of times does not yield a stationary process. Furthermore, this also implies that if bubbles do not exist stock prices and dividends should be cointegrated (see Diba and Grossman, 1988a). Applying unit root and cointegration tests to stock prices and dividends, Diba and Grossman (1988a) do not find any evidence of a stock-price bubble in the data. A similar application of the testing procedure can be found in Hamilton and Whiteman (1985). Despite these results, there is one major problem concerning the correct interpretation when the null hypothesis is rejected. In this case, the correct interpretation should be that the test indicates the presence of anything nonstationary which may stem, for example, from other unobservable fundamentals and need not necessarily indicate a speculative bubble (see Gürkaynak, 2008).

An alternative bubble test, that does not only use the theoretical properties but also a parametric specification of the bubble process, is due to Wu (1995) and Wu (1997). Wu (1997) treats the bubble as a latent variable and formulates the log-linear approximation of the standard present-value model as a linear state-space model. Using the Kalman filter, he estimates the unobserved bubble process as the deviation of the logarithm of the present value.[3] Similar to Diba and Grossman (1988a), this test also assumes a first-order autoregressive process. However, the advantage of this test is that the existence of a speculative bubble is directly formulated in the alternative hypothesis. Furthermore, the incorporation of the complete bubble specification enables to check the assumed bubble properties.[4] Using US stock data from 1871 to 1992, Wu (1997) finds that the estimated bubble component accounts for a substantial proportion of stock prices.

[3]For an application of this procedure to exchange rates see Wu (1995).

[4]One important drawback of this procedure is that the bubble component absorbs any misspecification of the model.

The article that mostly influenced the empirical literature on rational bubbles is due to Evans (1991). In this paper, Evans criticizes that the theoretical specifications of rational bubbles used in the tests so far do not appear to be empirically plausible since these bubble processes cannot collapse. Evans establishes a nonlinear bubble specification consistent with the theory of rational bubbles that exhibits a periodically collapsing behavior. Using simulated data, Evans demonstrates that the bubble test of Diba and Grossman (1988a) fails to detect rational bubbles if the stock price is driven by such a periodically collapsing bubble. This critique lead to a more recent strand of bubble tests being able to detect periodically collapsing bubbles. Hall et al. (1999) extends the Diba and Grossman's (1988a) testing methodology using a Markov-switching unit root test to account for periodically collapsing bubbles. Bohl (2003) applies a momentum threshold autoregessive model proposed by Enders and Siklos (2001) to account for periodically collapsing bubbles in the cointegration framework.

Al-Anaswah and Wilfling (2011) enrich the approach of Wu (1997) with Markov-switching elements. In their work the unobserved bubble component is represented by a first-order autoregressive bubble process whose autoregressive coefficients are allowed to switch between a surviving period and a collapsing state.

An alternative method of taking periodically collapsing rational bubbles into account consists in using sequential unit root tests. These tests, principally based on Diba and Grossman (1988a), use a recursive implementation of right-sided unit root tests. The procedures are able to estimate the origination and collapsing dates of the bubble process. Applications can be found in Phillips et al. (2011), Phillips and Yu (2011) and Homm and Breitung (2012).

Using simulated data and Evans' (1991) nonlinear bubble specification the more recent testing procedures appear to perform well in detecting periodically collapsing bubbles. When applied to real-world data, these tests provide evidence of bubbly periods in the following financial markets: in currency markets (see Hall et al., 1999), stock markets (see Al-Anaswah and Wilfling, 2011; Phillips et al., 2011; Homm and Breitung, 2012), and house, oil and commodity markets (see Phillips and Yu, 2011).

Finally, it remains to mention the empirical literature on intrinsic rational bubbles. Based on their theoretical model, Froot and Obstfeld (1991) derive a nonlinear relationship between the dividend-price ratio and dividends. They conclude that, in the absence of a bubble, stock prices and dividends should be cointegrated with a certain cointegration parameter. Applying a cointegration test on prices and dividends and performing direct estimation of the relationship between the dividend-price ratio and dividends for US stock-market data, Froot and Obstfeld reject the *no bubble* hypothesis and conclude that intrinsic bubbles are empirically relevant. For additional empirical work on intrinsic bubbles see Ma and Kanas (2004) and Driffill and Sola (1998).

Chapter 3

Estimation of periodically collapsing bubbles

3.1 Introduction

In the literature the most famous econometric specification for rational bubbles is the periodically collapsing bubble model proposed by Evans (1991). This stochastic and nonlinear model takes many theoretical properties of speculative bubbles into account and has decisively influenced the empirical bubble literature. In line with the literature on identification tests, the Evans bubble is used as the data generating process in simulation studies when examining the power of the test procedure (see, among others, Van Norden and Vigfusson, 1998; Hall et al., 1999; Bohl, 2003; Al-Anaswah and Wilfling, 2011; Phillips et al., 2011; Homm and Breitung, 2012). Although used in simulation studies, the Evans bubble is not part of the economic model. A first attempt to incorporate the Evans bubble into an economic framework is the paper of Brooks and Katsaris (2005). The authors propose a three-regime speculative behavior model in order to describe the dynamics of stock-market returns. The authors reproduce the general dynamics of the Evans bubble specification via a three regime-switching model and derive a linear switching regression model of

gross returns. However, in order to estimate this regression they construct bubble values by the difference between the current stock price and the fundamental value calculated by the model of Campbell and Shiller (1987).

Even though this work implies a periodically collapsing bubble specification close to the Evans model, the bubble component is simply estimated by the deviation of the current stock price and its fundamental value. The aim of this chapter is (a) to establish an economic model of the stock price that explicitly implies the structure of a periodically collapsing Evans bubble, and (b) to estimate the econometric structure of the Evans bubble from the data. To this end we treat the speculative bubble as a latent variable as in Wu (1997). Our innovation is that this latent variable is described by the nonlinear bubble model proposed by Evans (1991). For this purpose, we present a nonlinear state-space model that consists of a standard fundamental model of the stock price and the periodically collapsing Evans bubble. Based on this model we estimate the latent Evans bubble process from stock prices and dividends. This direct application of the nonlinear state-space model is a further innovation of this work. In contrast to the vast empirical literature on rational bubbles, we do not log-linearize the model so that the filtered bubble process can be interpreted unambiguously.

In order to estimate this nonlinear latent bubble process we use sequential Monte Carlo methods, namely the so-called particle filters. The first well functioning particle filter is due to Gordon et al. (1993) and has been mainly used in engineering. An overview can be found in Doucet et al. (2001) and in the survey of Creal (2012). The particle filter was introduced into economics by the paper of Kim et al. (1998) and has become increasingly popular in recent years (e.g. Fernández-Villaverde and Rubio-Ramírez, 2007; Kim and Stoffer, 2008; Duan and Fulop, 2009; Pitt et al.,

2014). A preliminary condition for applying the particle filter is that the model parameters are known. Especially for economic models and their empirical verification, however, the parameters are unknown and have to be estimated from the observed data. In this work we use a particle based approach of the Expectation Maximization (EM) algorithm developed by Schön et al. (2011) to identify the nonlinear state-space model.

The remainder of this chapter is organized as follows. Section 3.2 describes the economic specification of the nonlinear state-space model. Sections 3.3 and 3.4 review the estimation techniques used for the identification of the nonlinear state-space model. In Section 3.5 the econometric method is applied to artificial data and to a real-world data set consisting of prices and dividends of four major stock-price indices. Section 3.6 offers some concluding comments.

3.2 Model specification

In what follows we consider that the stock price is explained by the linear present-value model as described in Chapter 2.2 so that from Eq. (2.2) and Eq. (2.5) the stock price at date t is given by

$$P_t = \sum_{i=1}^{\infty} \left(\frac{1}{1+r} \right)^i \cdot E_t(D_{t+i}) + B_t.$$

To obtain a closed-form solution for the price at date t we need an assumption about the evolution of the future dividend payments. In the classical *Gordon growth model* (see Gordon, 1959) the dividends are expected to grow at a constant rate. Two alternative dividend models are the simple random walk and the random walk with drift which are standard assumptions in classical consumption-based asset pricing

models like the *Lucas Tree model* (see Lucas, 1978) and the model of Barro (2006). A common feature of these models is that the fundamental stock price can ultimately be written as any constant multiplied by the current level of dividend payments.[1]

In this paper, we assume that the price is simply given by

$$P_t = \phi \cdot D_t + B_t + \varepsilon_t, \tag{3.1}$$

where P_t is the real stock price at date t, D_t the real dividend payment between $t-1$ and t and B_t the latent bubble component. The error term ε_t is assumed to be Gaussian white-noise with variance σ_ε^2 reflecting all fundamentals that are not captured by dividends. Consequently, the fundamental value of the stock price is defined by $\phi \cdot D_t + \varepsilon_t$.

Since the rational bubble component B_t in Eq. (3.1) is not observable, we have to specify it more precisely. Wu (1997) assumes this bubble component to be a simple linear process like in Eq. (2.6). Following Evans' (1991) critique, such bubbles do not appear to be empirically plausible as they cannot collapse. Al-Anaswah and Wilfling (2011) consider a two-regime Markov-switching version of the bubble in Eq. (2.6) in order to account for a collapsing behavior. However, in this econometric specification negative bubbles cannot be excluded, whereas negative bubbles cannot occur in the economic rational-expectation model.

One innovation of this paper is that we explicitly consider the nonlinear bubble specification proposed by Evans (1991). Up to now, this model has been widely used in the literature since it takes into account many theoretical and empirical charac-

[1] The resulting fundamental stock price primarily differs in the structure of the constant. In the *Gordon growth model* the constant consists of the (real) rate of return and the dividend growth rate, whereas in the *Lucas Tree model* the constant equals a term including the mean and the variance of the dividend process, the time preference rate and relative risk aversion.

teristics of a speculative bubble. Furthermore, it satisfies the discounted martingale property so it is a solution to the Euler equation.

Defining the discount factor $\psi = (1+r)^{-1}$, we write the Evans bubble in the form

$$B_t = \begin{cases} \frac{1}{\psi} B_{t-1} u_t & , \text{ if } B_{t-1} \leq \tau \\ [\kappa + \frac{1}{\pi\psi}\Big(B_{t-1} - \kappa\psi\Big)\nu_t] u_t & , \text{ if } B_{t-1} > \tau \end{cases} , \tag{3.2}$$

where κ and τ are real constants such that $0 < \kappa < (1+r)\tau$. $\{u_t\}_{t=1}^{\infty}$ is an exogenous process of i.i.d. random variables with $u_t > 0$ and $E_{t-1}(u_t) = 1$ for all t. As in Evans (1991), we explicitly assume the variables $\{u_t\}$ to be lognormally (LN) distributed and scaled to have unit mean, i.e. we assume $u_t = \exp(y_t - \iota^2/2)$ with $\{y_t\}_{t=1}^{\infty}$ being i.i.d. $N(0, \iota^2)$.[2] $\{\nu_t\}_{t=1}^{\infty}$ is an exogenous i.i.d. Bernoulli process independent of $\{u_t\}_{t=1}^{\infty}$ with $\Pr(\nu_t = 1) = \pi$ and $\Pr(\nu_t = 0) = 1 - \pi$ for $0 < \pi \leq 1$. The event $\{\nu_t = 1\}$ means that the bubble will continue to grow, whereas the bubble bursts in case of $\{\nu_t = 0\}$.

It is instructive to note that the Evans bubble (3.2) has two different rates of growth. For $B_{t-1} \leq \tau$ the bubble grows at the mean rate $\frac{1}{\psi}$. In case of $B_{t-1} > \tau$ the bubble grows at the faster rate $\frac{1}{\pi\psi}$, but collapses with probability $1 - \pi$ per period. When the bubble collapses, it falls back to the mean value κ and the process recommences.

The economic model consists of two components, namely the price equation (3.1) and the bubble equation (3.2). Owing to Eq. (3.1) the stock price depends on observable dividend payments and an unobservable bubble component that is described by the nonlinear process from Eq. (3.2). As a result, we can interpret this two-equation system as a nonlinear state-space model.

[2] In other words, $\{u_t\}$ represents an i.i.d. distributed lognormal process with $u_t \sim LN(\frac{-\iota^2}{2}, \iota^2)$.

The direct application of the nonlinear state-space model is a further innovation of this paper. We use a clear-cut relationship between the stock price, its fundamental value and the bubble component that follows directly from the standard present-value model. Previous empirical literature on rational bubbles (*inter alia* Diba and Grossman, 1988a; Wu, 1997; Phillips et al., 2011; Al-Anaswah and Wilfling, 2011) uses the log-linear approximation of the standard present-value model as proposed by Campbell and Shiller (1988). However, the log-linearization of the model dissolves the relationship between the economic variables in Eq. (2.5) and the bubble component B_t is no longer defined as the deviation of the stock price from its fundamental value, but rather as a fundamental value-to-price ratio. In our paper, we circumvent these problems and can thus interpret the estimated bubble process unambiguously. To estimate our nonlinear state-space representation we use particle-filter methods which we describe below.

3.3 State-space estimation

In this section we introduce the particle-filter and particle-smoother approach. These sequential Bayesian Monte Carlo methods enable us to estimate the unobserved state variable, the bubble component, from a given nonlinear and non-Gaussian state-space model. Furthermore, we need these methods for parameter estimation. This section closely follows Schön et al. (2011) and Creal (2012).

3.3.1 Particle filter

We consider the general nonlinear state-space model characterized by the observation equation

$$y_t = m_t(x_t, o_t, \zeta_t) \tag{3.3}$$

and the state equation

$$x_t = h_t(x_{t-1}, o_t, \eta_t). \tag{3.4}$$

In Eq. (3.3) y_t represents the observable variable and in Eq. (3.4) x_t represents the state variable for $t = 1, ..., T$.[3] The variable o_t denotes an observable input variable. $\{\zeta_t\}_{t=1}^T$ and $\{\eta_t\}_{t=1}^T$ are mutually independent and i.i.d. noise processes with known densities while m_t and h_t are nonlinear functions of a given form. Thus, the distributions of the observation equation (3.3) and the distribution of the state equation (3.4) are given by $p(y_t \mid x_t; \boldsymbol{\theta})$ and $p(x_t \mid x_{t-1}; \boldsymbol{\theta})$, respectively.[4] These distributions depend on the parameter vector $\boldsymbol{\theta}$ which has to be estimated from all observations $y_{1:T}$.[5]

Since the states $x_{0:T} = (x_0, x_1, ..., x_T)$ are unknown they have to be estimated by means of all available data $y_{1:T} = (y_1, y_2, ..., y_T)$ and the structure of the underlying state-space model. To this end, it is appropriate to consider the joint conditional probability distribution $p(x_{0:T} \mid y_{1:T}; \boldsymbol{\theta})$, i.e. the density of the unknown states given the observed data. Once this distribution is determined, it is possible to estimate the sequence of the state variable (see Simon, 2006, pp. 462-466).

Following Creal (2012) and determining the expectation by $E(\cdot)$, we have

$$E(x_{0:T}) = \int x_{0:T} p(x_{0:T} \mid y_{1:T}; \boldsymbol{\theta}) dx_{0:T} \approx \frac{1}{N} \sum_{i=1}^{N} x_{0:T}^{(i)}. \tag{3.5}$$

By the law of large numbers a standard Monte Carlo estimator of the unknown states simply consists of the average of the N paths $\{x_{0:T}^{(i)}\}_{i=1}^N$ drawn from the target

[3] These random variables may be continuous, discrete or a combination of both. Furthermore, x_t and y_t may be vectors of more than one state and observable variable (see Creal, 2012). Here we consider a nonlinear state-space model consisting of one state and one observable variable.

[4] For the sake of simplicity we neglect the input variable o_t.

[5] We present the estimation procedure in Section 3.4. At the moment, we assume the parameter vector $\boldsymbol{\theta}$ to be known.

distribution $p(x_{0:T} \mid y_{1:T}; \boldsymbol{\theta})$, which justifies the approximation in Eq. (3.5). However, we generally do not know the target distribution rendering sampling from it impossible. To circumvent this problem, we use importance sampling. In doing so, we choose a proposal or importance distribution $g_{0:T}(x_{0:T} \mid y_{1:T}; \boldsymbol{\varphi})$ depending on the parameter vector $\boldsymbol{\varphi}$ from which it is possible to sample.[6] Thus, we can rewrite Eq. (3.5) as

$$E(x_{0:T}) = \int x_{0:T} \frac{p(x_{0:T} \mid y_{1:T}; \boldsymbol{\theta})}{g_{0:T}(x_{0:T} \mid y_{1:T}; \boldsymbol{\varphi})} g_{0:T}(x_{0:T} \mid y_{1:T}; \boldsymbol{\varphi}) dx_{0:T}. \tag{3.6}$$

Now, we can simply compute the expectation value by a weighted average of the N paths $\{x_{0:T}^{(i)}\}_{i=1}^{N}$ drawn from our proposal distribution:

$$E(x_{0:T}) \approx \sum_{i=1}^{N} x_{0:T}^{(i)} \frac{w^{(i)}}{\sum_{j=1}^{N} w^{(j)}}, \tag{3.7}$$

with

$$w^{(i)} \propto \frac{p(x_{0:T}^{(i)} \mid y_{1:T}; \boldsymbol{\theta})}{g_{0:T}(x_{0:T}^{(i)} \mid y_{1:T}; \boldsymbol{\varphi})}, \tag{3.8}$$

where the weights $\{w^{(i)}\}_{i=1}^{N}$ define the relationship between the target and the proposal distribution and function as a correction for consciously drawing from the wrong distribution.

In our setting it is more appropriate to modify the above-described procedure in order to draw sequentially from a sequence of conditional distributions. To this end, the proposal distribution is factored into two parts and we obtain for any t

$$g_{0:t}(x_{0:t} \mid y_{1:t}; \boldsymbol{\varphi}) \equiv g_t(x_t \mid x_{0:t-1}, y_{1:t}; \boldsymbol{\varphi}) g_{0:t-1}(x_{0:t-1} \mid y_{1:t-1}; \boldsymbol{\varphi}). \tag{3.9}$$

[6] From a technical point of view, it is important that the support of proposal distribution covers the support of the target distribution.

Thus, at any point in time t, we only draw a new set of N values $\{x_t^{(i)}\}_{i=1}^N$ from the first part of the proposal distribution in Eq. (3.9), $g_t(x_t \mid x_{0:t-1}, y_{1:t}; \boldsymbol{\varphi})$, and attach this set to the realized paths up to date $t-1$, so that the new paths in t equal $\{x_{0:t}^{(i)}\}_{i=1}^N = \{x_{0:t-1}^{(i)}, x_t^{(i)}\}_{i=1}^N$.[7] If we write the joint conditional probability distribution recursively, we obtain the weight for path i at time t as

$$\begin{aligned} w_t^{(i)} &= \frac{p(y_t \mid x_t^{(i)}; \boldsymbol{\theta}) p(x_t^{(i)} \mid x_{t-1}^{(i)}; \boldsymbol{\theta}) p(x_{0:t-1}^{(i)} \mid y_{1:t-1}; \boldsymbol{\theta})}{p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}) g_t(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t}; \boldsymbol{\varphi}) g_{0:t-1}(x_{0:t-1}^{(i)} \mid y_{1:t-1}; \boldsymbol{\varphi})} \\ &\propto w_{t-1}^{(i)} \frac{p(y_t \mid x_t^{(i)}; \boldsymbol{\theta}) p(x_t^{(i)} \mid x_{t-1}^{(i)}; \boldsymbol{\theta})}{g_t(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t}; \boldsymbol{\varphi})} \\ &\propto w_{t-1}^{(i)} \hat{w}_t^{(i)}, \end{aligned} \tag{3.10}$$

where

$$\hat{w}_t^{(i)} = \frac{p(y_t \mid x_t^{(i)}; \boldsymbol{\theta}) p(x_t^{(i)} \mid x_{t-1}^{(i)}; \boldsymbol{\theta})}{g_t(x_t^{(i)} \mid x_{0:t-1}^{(i)}, y_{1:t}; \boldsymbol{\varphi})}. \tag{3.11}$$

The advantage of this method is that at each date t we only have to calculate the ratio in Eq. (3.11) while the previous weights can be updated by Eq. (3.10). Instead of recomputing the entire expressions from Eqs. (3.7) and (3.8) we can simply update our estimation when a new observation y_{t+1} becomes available. Thus, at the time step t, we obtain N paths of the state variable of length t and N corresponding weights $\{x_{0:t-1}^{(i)}, x_t^{(i)}, w_t^{(i)}\}_{i=1}^N$. This method is known as sequential importance sampling and the draws are called particles.

One potential problem occurring in sequential importance sampling is that after a certain time period one particle's normalized weight converges to 1 while the rest are converging to 0. As a result, the estimate of the unobserved state consists of a single draw. This phenomenon is known as *weight degeneracy* and can be circumvented by an additional resampling step at each date t. Following Gordon et al. (1993), at each

[7] Note that the function g_t can change over time.

time step t we draw a new set of values $\{\tilde{x}_t^{(i)}\}_{i=1}^N$ from the existing sample $\{x_t^{(i)}\}_{i=1}^N$ in proportion to their normalized weights $\{w_t^{(i)}/\sum_{j=1}^N w_t^{(j)}\}_{i=1}^N$. After resampling, all weights are set to $w_t^{(i)} = 1/N$ for $i = 1, ..., N$. In the literature on particle-filter methods, various resampling algorithms are available (see Douc and Cappé, 2005 for a comparison of these approaches). In this paper we use the systematic resampling algorithm. The combination of sequential importance sampling and resampling is the basic particle-filtering approach.[8]

As a result, the particle-filtering algorithm establishes an approximation of the filtered density $p(x_t \mid y_{1:t}; \boldsymbol{\theta})$

$$p(x_t \mid y_{1:t}; \boldsymbol{\theta}) \approx \sum_{i=1}^{N} \tilde{w}_t^{(i)} \delta(x_t - x_t^{(i)}), \tag{3.12}$$

where δ denotes the Dirac measure and $\tilde{w}_t^{(i)} = w_t^{(i)}/\sum_{j=1}^N w_t^{(j)}$. Furthermore, on the analogy of Eqs. (3.5) and (3.7), we can calculate the Monte Carlo estimator of the expected value of the unobserved state variable at time t by

$$E(x_t) = \int x_t p(x_t \mid y_{1:t}; \boldsymbol{\theta}) dx_t \approx \sum_{i=1}^{N} x_t^{(i)} \tilde{w}_t^{(i)}. \tag{3.13}$$

Alternatively, after the resampling step, the estimator can be written as

$$E(x_t) \approx \frac{1}{N} \sum_{i=1}^{N} \tilde{x}_t^{(i)}. \tag{3.14}$$

Finally, we need a suitable proposal distribution for the implementation of the particle filter. A simple and useful way is to choose the density of the state equation, i.e. $g_t(x_t \mid x_{0:t-1}, y_{1:t}; \boldsymbol{\varphi}) = p(x_t \mid x_{t-1}; \boldsymbol{\theta})$. Since this density is known, we can easily sample from it. The advantage is that the weights in Eq. (3.11) are then only

[8] There are several types of particle filters which only differ in their choice of the proposal distribution $g_t(x_t \mid x_{0:t-1}, y_{1:t}; \boldsymbol{\varphi})$ and the resampling algorithm. For an overview see Creal (2012).

defined by the density of the observation equation $p(y_t \mid x_t; \boldsymbol{\theta})$ which is also known. This type of the particle filter corresponds to the original particle filter developed by Gordon et al. (1993). It is also called bootstrap filter and will also be used below.

3.3.2 Particle smoother

Based on all available observations up to time T, it is possible to improve all state estimates by smoothing. For this purpose we determine the smoothed density $p(x_t \mid y_{1:T}; \boldsymbol{\theta})$ an approximation of which can be obtained by a particle smoother. Our smoothing method described below closely follows Schön et al. (2011) and is equivalent to the reweighting particle smoother proposed by Hürzeler and Künsch (1998) and Doucet et al. (2000). This method uses all available observations to improve the particle weights and results in a more exact state estimation. We use this smoothing method since it is also part of the parameter estimation method proposed by Schön et al. (2011) in Section 3.4. However, several alternative particle-smoother methods are available in literature (see Godsill et al., 2004 and Briers et al., 2010).

Using the law of total probability, we can write the required smoothed density as

$$p(x_t \mid y_{1:T}; \boldsymbol{\theta}) = \int p(x_t \mid x_{t+1}, y_{1:t}; \boldsymbol{\theta}) p(x_{t+1} \mid y_{1:T}; \boldsymbol{\theta}) dx_{t+1}. \tag{3.15}$$

Invoking the definition of conditional probability and Bayes' rule, we can express Eq. (3.15) as

$$p(x_t \mid y_{1:T}; \boldsymbol{\theta}) = p(x_t \mid y_{1:t}; \boldsymbol{\theta}) \int \frac{p(x_{t+1} \mid x_t; \boldsymbol{\theta}) p(x_{t+1} \mid y_{1:T}; \boldsymbol{\theta})}{p(x_{t+1} \mid y_{1:t}; \boldsymbol{\theta})} dx_{t+1}, \tag{3.16}$$

and, via the law of total probability, the denominator of the integral may be represented as

$$p(x_{t+1} \mid y_{1:t}; \boldsymbol{\theta}) = \int p(x_{t+1} \mid x_t; \boldsymbol{\theta}) p(x_t \mid y_{1:t}; \boldsymbol{\theta}) dx_t. \tag{3.17}$$

Now, we can express the smoothed density in terms of the filtered density, Eq. (3.12), the density of the state equation and the smoothed density at date $t+1$:

$$p(x_t \mid y_{1:T}; \boldsymbol{\theta}) = p(x_t \mid y_{1:t}; \boldsymbol{\theta}) \int \frac{p(x_{t+1} \mid x_t; \boldsymbol{\theta}) p(x_{t+1} \mid y_{1:T}; \boldsymbol{\theta})}{\int p(x_{t+1} \mid x_t; \boldsymbol{\theta}) p(x_t \mid y_{1:t}; \boldsymbol{\theta}) dx_t} dx_{t+1}. \tag{3.18}$$

Finally, using the Dirac delta approximation, we rewrite this equation as

$$p(x_t \mid y_{1:T}; \boldsymbol{\theta}) \approx \sum_{i=1}^{N} \tilde{w}_{t|1:T}^{(i)} \delta(x_t - x_t^{(i)}), \tag{3.19}$$

where

$$\tilde{w}_{t|1:T}^{(i)} = \tilde{w}_t^{(i)} \sum_{k=1}^{N} \frac{\tilde{w}_{t+1|1:T}^{(k)} p(x_{t+1}^{(k)} \mid x_t^{(i)}; \boldsymbol{\theta})}{\sum_{j=1}^{N} \tilde{w}_t^{(j)} p(x_{t+1}^{(k)} \mid x_t^{(j)}; \boldsymbol{\theta})}. \tag{3.20}$$

To apply this smoothing algorithm, we need the particles and the corresponding weights $\{x_t^{(i)}, w_t^{(i)}\}_{i=1}^N$ at each date t from the sequential importance sampling. At $t = T$ we have $\tilde{w}_{T|1:T}^{(i)} = \tilde{w}_T^{(i)}$ and, based on the results of the particle filter and the dynamics of the model in Eq. (3.4), the smoothed density can be calculated recursively.[9]

Based on the particles $\{x_t^{(i)}\}_{i=1}^N$ from the sequential importance sampling and the new smoothed weights $\{\tilde{w}_{t|1:T}^{(i)}\}_{i=1}^N$, we can now estimate the smoothed states at each date t, given all observations up to time T, $\{x_{t|1:T}^{(i)}\}_{i=1}^N$. In analogy to Eq. (3.13), we compute the expectation value of the unobserved state variable at time t as

$$E(x_{t|1:T}) \approx \sum_{i=1}^{N} x_t^{(i)} \tilde{w}_{t|1:T}^{(i)}. \tag{3.21}$$

[9] For a detailed derivation see Schön et al. (2011), pp. 43-44.

3.4 Parameter estimation

3.4.1 Parameter estimation by maximum likelihood

In most empirical applications the parameter vector $\boldsymbol{\theta}$ is unknown and has to be estimated from the data which is typically executed by maximum likelihood techniques. However, for nonlinear state-space models parameter maximum likelihood estimation is not straightforward. If we use particle-filter methods for approximating the likelihood function, the resulting function is not continuous in the parameter vector (see Hürzeler and Künsch, 2001 and Creal, 2012). This problem is caused by the additional resampling step at each iteration. In the resampling step we use a discrete cumulative distribution function defined by the particle weights to generate a set of resampled particles. Even if we use common random numbers a small change in the parameter vector will change the particle weights and, thus, potentially generate different resampled particles (see Pitt, 2002 and Kantas et al., 2009). Consequently, the surface of the likelihood function is rough and the maximization of the likelihood function using gradient-based optimizer becomes difficult or even impossible (see Pitt, 2002).

To overcome this problem alternative approaches to maximize the likelihood function like stochastic gradient-based methods (see Poyiadjis et al., 2005), recursive maximum likelihood methods (see Doucet and Tadic, 2003) and smooth particle-filter methods may be used (see Pitt, 2002 and Malik and Pitt, 2011).[10] A frequently applied method to overcome the discontinuity of the likelihood function is the Expectation Maximization algorithm for which several variants are available in nonlinear and non-Gaussian state-space frameworks (see Kim and Stoffer, 2008; Olsson et al.,

[10] Alternatively, Markov chain Monte Carlo methods can also be used for parameter estimation (see Andrieu et al., 2010).

2008 and Schön et al., 2011, *inter alia*). In this paper we use the EM algorithm established by Schön et al. (2011) for parameter estimation which we now briefly review.

3.4.2 Maximum likelihood estimation via the EM algorithm

The Expectation Maximization (EM) algorithm constitutes an iterative procedure for computing the maximum likelihood estimator in which each iteration consists of an expectation plus a maximization step. This algorithm goes back to Dempster et al. (1977) and has been applied to a variety of incomplete-data problems (see McLachlan and Krishnan, 2007). The following review is close to the description in Schön et al. (2011).

Let $y_{1:T}$ denote the observable data and $\log L_{\boldsymbol{\theta}}(y_{1:T}) = \log p(y_{1:T}; \boldsymbol{\theta})$ the log-likelihood function of all observations given the parameter vector $\boldsymbol{\theta}$. The EM algorithm computes the parameter vector $\boldsymbol{\theta}$ at iteration k such that the observed-data log-likelihood function exceeds the one-step-before log-likelihood function, i.e. $\log L_{\boldsymbol{\theta}_k}(y_{1:T}) > \log L_{\boldsymbol{\theta}_{k-1}}(y_{1:T})$, where it is assumed that the data set is incomplete (see Dempster et al., 1977). Following Schön et al. (2011), we denote these missing data by $x_{1:T}$ and consider the complete-data log-likelihood function

$$\log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T}) = \log p(y_{1:T}, x_{1:T}; \boldsymbol{\theta}).^{11} \tag{3.22}$$

In general, this joint log-likelihood function has a *nicer* shape and maximization becomes more tractable (see McLachlan and Krishnan, 2007). Since we can interpret our nonlinear and non-Gaussian state-space model as an incomplete-data

[11] In subsequent sections, the missing data set will be replaced with the state vector.

problem and due to the problems resulting from discontinuous likelihood functions, this method seems to be the method of choice in our context.

Owing to the fact that missing data are not observable, the joint log-likelihood function is approximated by its expectation value conditional on the observed values plus an assumption on the true parameter vector $\boldsymbol{\theta}_k$:

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) &\triangleq E_{\boldsymbol{\theta}_k}(\log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T}) \mid y_{1:T}) \\ &= \int \log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T}) p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}_k) dx_{1:T}. \end{aligned} \tag{3.23}$$

Next, rewriting Eq. (3.22) (by using the definition of conditional probability) as

$$\log p(y_{1:T}, x_{1:T}; \boldsymbol{\theta}) = \log p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}) + \log p(y_{1:T}; \boldsymbol{\theta}), \tag{3.24}$$

and taking the conditional expectation $E_{\boldsymbol{\theta}_k}(\cdot \mid y_{1:T})$ of this equation, we obtain

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) = \log L_{\boldsymbol{\theta}}(y_{1:T}) + \int \log p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}) p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}_k) dx_{1:T}. \tag{3.25}$$

Now, in view of Eq. (3.25), the following relationship obtains:

$$\begin{aligned} \log L_{\boldsymbol{\theta}}(y_{1:T}) - \log L_{\boldsymbol{\theta}_k}(y_{1:T}) &= Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) - Q(\boldsymbol{\theta}_k, \boldsymbol{\theta}_k) \\ &\quad + \int \log \frac{p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}_k)}{p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta})} p(x_{1:T} \mid y_{1:T}; \boldsymbol{\theta}_k) dx_{1:T}. \end{aligned} \tag{3.26}$$

The integral on the right hand side equals the Kullback-Leibler divergence which is always non-negative.[12] Thus,

$$\log L_{\boldsymbol{\theta}}(y_{1:T}) - \log L_{\boldsymbol{\theta}_k}(y_{1:T}) \geq Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) - Q(\boldsymbol{\theta}_k, \boldsymbol{\theta}_k), \tag{3.27}$$

[12] For details see Schön et al. (2011), p. 41.

which emphasizes the basic idea behind the EM algorithm. At each iteration k, the parameter vector $\boldsymbol{\theta}$ is chosen such that $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) > Q(\boldsymbol{\theta}_k, \boldsymbol{\theta}_k)$ and by Eq. (3.27) implying $\log L_{\boldsymbol{\theta}}(y_{1:T}) > \log L_{\boldsymbol{\theta}_k}(y_{1:T})$. Thus, the EM algorithm becomes a two-step procedure after starting with $k = 0$ and initiating the algorithm with a starting vector $\boldsymbol{\theta}_{k=0}$: In the expectation step, we calculate the Q-function given the vector $\boldsymbol{\theta}_k$, i.e. $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$. In the following maximization step, we maximize this Q-function subject to $\boldsymbol{\theta}$. As long as this procedure does not converge, k is updated to $k + 1$ and a new iteration is conducted.

For the application to nonlinear state-space models it is necessary to compute the approximated joint log-likelihood function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ from Eq. (3.23). In general nonlinear state-space models, as defined in Eqs. (3.3) and (3.4), we may use Bayes' rule and the Markov nature of the state equation (3.4) to write the joint log-likelihood function as

$$\begin{aligned} \log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T}) &= \log p(y_{1:T} \mid x_{1:T}; \boldsymbol{\theta}) + \log p(x_{1:T}; \boldsymbol{\theta}) \\ &= \log p(x_1; \boldsymbol{\theta}) + \sum_{t=1}^{T-1} \log p(x_{t+1} \mid x_t; \boldsymbol{\theta}) \\ &\quad + \sum_{t=1}^{T} \log p(y_t \mid x_t; \boldsymbol{\theta}). \end{aligned} \tag{3.28}$$

In analogy to Eq. (3.23), we may take the conditional expectation $E_{\boldsymbol{\theta}_k}(\cdot \mid y_{1:T})$ to obtain

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k) &= \int \log p(x_1; \boldsymbol{\theta}) p(x_1 \mid y_{1:T}; \boldsymbol{\theta}_k) dx_1 \\ &\quad + \sum_{t=1}^{T-1} \int \int \log p(x_{t+1} \mid x_t; \boldsymbol{\theta}) p(x_{t+1}, x_t \mid y_{1:T}; \boldsymbol{\theta}_k) dx_t dx_{t+1} \\ &\quad + \sum_{t=1}^{T} \int \log p(y_t \mid x_t; \boldsymbol{\theta}) p(x_t \mid y_{1:T}; \boldsymbol{\theta}_k) dx_t. \end{aligned} \tag{3.29}$$

Finally, for computing of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ we need the densities of the observation and state equation plus the densities $p(x_t \mid y_{1:T}; \boldsymbol{\theta}_k)$ and $p(x_{t+1}, x_t \mid y_{1:T}; \boldsymbol{\theta}_k)$, which are a by-product of the particle-smoother method.

3.4.3 EM algorithm

Based on the results above, we may approximate the Q-function, $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$, by using the particle-filtering methods from Section 3.3.1 and 3.3.2. We finally implement the EM algorithm as follows. In the expectation step at iteration k, we use the results from Eqs. (3.19) and (3.20) in conjunction with Eq. (3.12) to approximate the Q-function by[13]

$$\begin{aligned}\hat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}_k) &= \sum_{i=1}^{N} \tilde{w}_{1|1:T}^{(i)} \log p(x_1^{(i)}; \boldsymbol{\theta}) \\ &\quad + \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} \tilde{w}_{t|1:T}^{(ij)} \log p(x_{t+1}^{(j)} \mid x_t^{(i)}; \boldsymbol{\theta}) \\ &\quad + \sum_{t=1}^{T} \sum_{i=1}^{N} \tilde{w}_{t|1:T}^{(i)} \log p(y_t \mid x_t^{(i)}; \boldsymbol{\theta}), \end{aligned} \tag{3.30}$$

where

$$\tilde{w}_{t|1:T}^{(ij)} = \frac{\tilde{w}_t^{(i)} \tilde{w}_{t+1|1:T}^{(j)} p(x_{t+1}^{(j)} \mid x_t^{(i)}; \boldsymbol{\theta}_k)}{\sum_{l=1}^{N} \tilde{w}_t^{(l)} p(x_{t+1}^{(j)} \mid x_t^{(l)}; \boldsymbol{\theta}_k)}, \tag{3.31}$$

and $\{x_t^{(i)}\}_{i=1}^N$ are the particles from the sequential importance sampling.

In the following maximization step we optimize this function subject to $\boldsymbol{\theta}$ and obtain $\boldsymbol{\theta}_{k+1}$. If the non-convergence criterion $\hat{Q}(\boldsymbol{\theta}_{k+1}, \boldsymbol{\theta}_k) - \hat{Q}(\boldsymbol{\theta}_k, \boldsymbol{\theta}_k) > c$ is reached for any $c > 0$, k is updated to $k + 1$ and a new iteration is conducted.[14] Otherwise, the algorithm terminates and $\boldsymbol{\theta}_k$ is the desired maximum likelihood estimator $\hat{\boldsymbol{\theta}}$.

[13] For a detailed derivation see Schön et al. (2011).

[14] The convergence properties of the EM algorithm via particle-filtering methods are the same as those of general EM methods. It can be shown that $\hat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ is an arbitrarily accurate approximation to $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ (see Hu et al., 2008 and Schön et al., 2011).

3.4.4 Standard errors

One drawback in the parameter estimation via the EM algorithm is that it does not automatically provide estimates of the covariance matrix of the maximum likelihood estimators. There are, however, several methods to be executed after estimation providing estimates of the information matrix under the EM algorithm (see McLachlan and Krishnan, 2007). In what follows we use a stable estimator of the information matrix proposed by Duan and Fulop (2011) to calculate the standard errors of the maximum likelihood estimates. We briefly outline this approach and adapt it to our empirical problem.

Under the well-known regularity conditions, an important property of the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ is its asymptotic normality, i.e.

$$\hat{\boldsymbol{\theta}} \overset{a}{\sim} N\Big(\boldsymbol{\theta}^*, I(\boldsymbol{\theta}^*)^{-1}\Big), \tag{3.32}$$

where $\boldsymbol{\theta}^*$ denotes the true parameter vector. $I(\boldsymbol{\theta}^*)$ represents the information matrix which can be expressed by the negative expected value of the Hessian of the log-likelihood function:

$$I(\boldsymbol{\theta}^*) = -E[H(\boldsymbol{\theta}^*)]. \tag{3.33}$$

The covariance matrix is therefore given by the inverse of the negative expected value of the Hessian (see Greene, 2012, p. 554). In context with the EM algorithm, the most widely used approach to calculate the covariance matrix is based on Eq. (3.33) and was proposed by Louis (1982). Following Louis (1982), the observed-data Hessian can be calculated using the joint log-likelihood function within the EM framework. However, one shortcoming of the method is that this empirical

Hessian may be unstable and negative definiteness cannot be guaranteed entailing a frequently encountered problem in practice.

Duan and Fulop (2011) present a more stable alternative based on the information-matrix inequality

$$Var(S(\boldsymbol{\theta}^*)) = E[S(\boldsymbol{\theta}^*)S(\boldsymbol{\theta}^*)'] = -E[H(\boldsymbol{\theta}^*)]. \tag{3.34}$$

Here, the variance of the score vector of the observed-data log-likelihood function is equal to the negative expected value of the Hessian (see Greene, 2012, p. 557). Using the prediction-error decomposition of the observed-data log-likelihood function, we find

$$\log L_{\boldsymbol{\theta}^*}(y_{1:T}) = \log p(y_1, ..., y_T; \boldsymbol{\theta}^*) = \sum_{t=1}^{T} \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*), \tag{3.35}$$

and the score $S(\boldsymbol{\theta}^*)$ of this log-likelihood is given by

$$S(\boldsymbol{\theta}^*) = \frac{\partial \log L_{\boldsymbol{\theta}^*}(y_{1:T})}{\partial \boldsymbol{\theta}} = \sum_{t=1}^{T} \frac{\partial \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}}. \tag{3.36}$$

Following Duan and Fulop (2011), we can then establish Eq. (3.34) as

$$\begin{aligned}
Var(S(\boldsymbol{\theta}^*)) &= E[S(\boldsymbol{\theta}^*)S(\boldsymbol{\theta}^*)'] \\
&= \sum_{t=1}^{T}\sum_{j=1}^{T} E\left[E\left(\frac{\partial \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}}\right.\right. \\
&\quad \left.\left. \times \frac{\partial \log p(y_j \mid y_{1:j-1}; \boldsymbol{\theta}^*)'}{\partial \boldsymbol{\theta}}\right) \middle| y_{1:t-1}\right] \\
&= \sum_{t=1}^{T} E\left[E\left(\frac{\partial \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}}\right.\right. \\
&\quad \left.\left. \times \frac{\partial \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*)'}{\partial \boldsymbol{\theta}}\right) \middle| y_{1:t-1}\right] \\
&= \sum_{t=1}^{T} E\left[-E\left(\frac{\partial^2 \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}^*)}{\partial \boldsymbol{\theta}^2}\right) \middle| y_{1:t-1}\right] \\
&= -E[H(\boldsymbol{\theta}^*)].
\end{aligned} \tag{3.37}$$

The idea pursued by Duan and Fulop (2011) now is to approximate the $Var(S(\boldsymbol{\theta}^*))$ by the smoothed individual scores of the joint log-likelihood function instead of using the scores of the observed-data log-likelihood from Eq. (3.36).

Louis (1982) shows that, for any $\boldsymbol{\theta}$, the score of the observed-data log-likelihood can be written as

$$S(\boldsymbol{\theta}) = E\left(\frac{\partial \log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T})}{\partial \boldsymbol{\theta}}\middle| y_{1:T}; \boldsymbol{\theta}\right), \tag{3.38}$$

and by definition of the joint log-likelihood function from Eq. (3.28) this score of the observed-data log-likelihood can be decomposed into the sum of smoothed individual scores:

$$\begin{aligned} S(\boldsymbol{\theta}) &= E\left(\frac{\partial \log L_{\boldsymbol{\theta}}(y_{1:T}, x_{1:T})}{\partial \boldsymbol{\theta}}\middle| y_{1:T}; \boldsymbol{\theta}\right) \\ &= E\left(\frac{\partial \log p(x_1; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\middle| y_{1:T}; \boldsymbol{\theta}\right) + \sum_{t=1}^{T-1} E\left(\frac{\partial \log p(x_{t+1} \mid x_t; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\middle| y_{1:T}; \boldsymbol{\theta}\right) \\ &\quad + \sum_{t=1}^{T} E\left(\frac{\partial \log p(y_t \mid x_t; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\middle| y_{1:T}; \boldsymbol{\theta}\right). \end{aligned} \tag{3.39}$$

Kim and Stoffer (2008) calculate this score function directly by the use of samples from the smoothed density, that is $x^{(i)}_{1:t|1:T}$ $(i = 1, ..., N)$. In what follows we use the definition of the approximated Q-function, that is the expected value of the joint log-likelihood function conditional on all observations and the parameter vector $\boldsymbol{\theta}$ (see Section 3.4.3). In this case we may estimate the score of the observed-data log-likelihood function by

$$\begin{aligned} \hat{S}(\boldsymbol{\theta}) &= \sum_{i=1}^{N} \tilde{w}^{(i)}_{1|1:T} \left(\frac{\partial \log p(x^{(i)}_1; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right) \\ &\quad + \sum_{t=1}^{T-1} \sum_{i=1}^{N} \sum_{j=1}^{N} \tilde{w}^{(ij)}_{t|1:T} \left(\frac{\partial \log p(x^{(j)}_{t+1} \mid x^{(i)}_t; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right) \end{aligned}$$

$$+\sum_{t=1}^{T}\sum_{i=1}^{N}\tilde{w}_{t|1:T}^{(i)}\left(\frac{\partial \log p(y_t \mid x_t^{(i)};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right). \tag{3.40}$$

However, since the smoothed individual scores do not constitute martingale differences, Duan and Fulop (2011) emphasize taking into account the dependence among lagged terms to approximate $Var(S(\boldsymbol{\theta}^*))$. If dependence becomes negligible beyond some lags, l, we can approximate $Var(S(\boldsymbol{\theta}^*))$ by the approach of Newey and West (1987):

$$Var(S(\boldsymbol{\theta}^*)) \approx A_0 + \sum_{j=1}^{l}\lambda(l)(A_j + A_j'), \tag{3.41}$$

where

$$A_j = \sum_{t=1}^{T-j} a_t(\hat{\boldsymbol{\theta}})a_{t+j}(\hat{\boldsymbol{\theta}})', \tag{3.42}$$

with

$$\begin{aligned} a_t(\hat{\boldsymbol{\theta}}) \;=\; & \sum_{i=1}^{N}\sum_{j=1}^{N}\tilde{w}_{t|1:T}^{(ij)}\left(\frac{\partial \log p(x_t^{(j)} \mid x_{t-1}^{(i)};\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}}\right) \\ & +\sum_{i=1}^{N}\tilde{w}_{t|1:T}^{(i)}\left(\frac{\partial \log p(y_t \mid x_t^{(i)};\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}}\right) \end{aligned} \tag{3.43}$$

and

$$\lambda(l) = 1 - \frac{j}{l+1}. \tag{3.44}$$

Finally, a positively semi-definite estimate of the covariance matrix can be calculated by the inverse of the approximation of $Var(S(\boldsymbol{\theta}^*))$ given in Eqs. (3.41) - (3.44).

3.5 Empirical application

In this section, we apply our economic model and the econometric methods to the data. To check the overall reliability of the technique, we first apply it to an artificial data set. In a second step, we use the nonlinear state-space representation and the

particle-filter methods to estimate the unobservable Evans-bubble process in a real-world data set covering stock prices and dividends.

3.5.1 Nonlinear state-space representation

In a preliminary step, it is necessary to combine the economic model with the particle-filtering approach described above. According to the economic specification from Section 3.2, we represent the stock price by Eq. (3.1) and the latent bubble process by Eq. (3.2). Furthermore, we need the distributions of the observation and the state equation, respectively.

The Gaussian distribution of the error term ε_t from Eq. (3.1) implies that the observation equation is also Gaussian:

$$p(P_t \mid B_t, D_t; \boldsymbol{\theta}) \sim N\Big((\phi D_t + B_t), \sigma_\varepsilon^2\Big). \tag{3.45}$$

The stock price P_t corresponds to the observable variable and the dividend payment D_t represents a further observable input variable. The bubble component B_t constitutes the state variable.

The conditional distribution of the state equation is given by the distribution of the Evans bubble. For $B_{t-1} \leq \tau$ the distribution is lognormal while for $B_{t-1} > \tau$ the distribution is a mixture of two lognormal distributions with mixing weights given

by the Bernoulli process. Overall, we write the distribution of the state equation in the form

$$p(B_t \mid B_{t-1}; \boldsymbol{\theta}) \sim \begin{cases} LN\left(\frac{-\iota^2}{2} + \log\left(\frac{1}{\psi} B_{t-1}\right), \iota^2\right) & \text{, if } B_{t-1} \leq \tau \\ \pi \cdot LN\left(\frac{-\iota^2}{2} + \log\left(\kappa + \frac{1}{\pi\psi}\left(B_{t-1} - \kappa\psi\right)\right), \iota^2\right) & \\ +(1-\pi) \cdot LN\left(\frac{-\iota^2}{2} + \log(\kappa), \iota^2\right) & \text{, if } B_{t-1} > \tau \end{cases}, \quad (3.46)$$

where the vector $\boldsymbol{\theta} = (\phi, \sigma_\varepsilon^2, \psi, \iota^2, \kappa, \pi, \tau)$ contains all model parameters.

Figure 3.1 about here

3.5.2 Artificial data

In order to generate an artificial data set of (real) stock prices and (real) dividends, we need realisations from the Evans-bubble process and a stochastic process governing the dividend payments. Our dividends are generated by a simple random walk while the values of the Evans bubble are generated by Eq. (3.2). Finally, the stock price is calculated by the price equation given in Eq. (3.1). The parameter vector is set to be $\boldsymbol{\theta} = (\phi = 50, \sigma_\varepsilon^2 = 1.2, \psi = 0.9804, \iota^2 = 0.001, \kappa = 1.1, \pi = 0.98, \tau = 2)$. In line with the rational-expectation model an at least infinitesimal small positive bubble must already be part of the stock price (see Section 2.1.1), so that we initiate the bubble process with the value $B_0 = 0.5$. Our data set consists of 250 observations, which equals an observation period of approximately 21 years based

on monthly data and contains two bubbles.[15] Figure 3.1 displays the simulated data of the economic model.

3.5.3 Estimation results

Table 3.1 displays the parameter estimates based on the maximum likelihood procedure via the particle based approach of the EM algorithm. Owing to the computational effort of this estimation method, we only used $N = 300$ particles in the estimation. For the convergence criterion we defined $c = 1/N \approx 0.0033$. However, if the algorithm did not converge, we terminated estimation after 500 iterations to save computation time.[16] We initiated the EM algorithm with the parameter vector $\boldsymbol{\theta_0} = (\phi = 20, \sigma_\varepsilon^2 = 0.5, \psi = 0.8, \iota^2 = 0.005, \kappa = 0.5, \pi = 0.8, \tau = 2)$ and numerically maximized the Q-function from Eqs. (3.30) and (3.31) by the FMINCON module of the software package MATLAB.[17] The standard errors were computed by the stable estimator of the information matrix established by Duan and Fulop (2011) as described in Section 3.4.4.[18]

Table 3.1 about here

Owing to an identification problem, we refrained from estimating the parameter τ and used the true value $\tau = 2$ instead. Due to our convergence criterion the

[15] The parameter values of the Evans process are different from those given in the original paper by Evans (1991). We choose this parameterization in order to generate only a few big bubbles in the data. In line with the history of financial markets this appears to be a realistic assumption for the length of our data (monthly observations).

[16] The convergence of the EM algorithm depends on the quality of the approximation of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$. Following Schön et al. (2011), $\hat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ is an arbitrarily accurate approximation if $N \to \infty$. We defined c depending on N to account for a less accurate approximation because of a small number of particles. However, even if a small number of particles is used, this procedure is appropriate to generate reasonable estimates (see Schön et al., 2011).

[17] The choice of the initial parameter vector is crucial to the convergence speed of the EM algorithm. We set the starting values arbitrarily, but sufficiently far away from the true values.

[18] We used 500 particles for the computation of the standard errors and considered $l = 15$ lags.

algorithm stopped after $k = 125$ iterations. All parameter estimates are significant at conventional levels. The parameters ϕ, σ_ε^2, ψ and π are rather precisely estimated, while the parameter estimates of ι^2 and κ do not appear to be very close. From an economic point of view the most interesting parameters are the discount factor ψ, the bubble survival probability π and the dividend parameter ϕ. All these parameters are well estimated.

Figure 3.2 about here

To illustrate the above-described discontinuity problem of the likelihood function and to show the advantage of the EM algorithm, we compare the several dimensions of the true observed-data log-likelihood function, $\log L_{\boldsymbol{\theta}_k}(y_{1:T})$, with the corresponding dimensions of the approximated Q-function, $\hat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$, after $k = 125$ iterations.[19] Figure 3.2 displays that for some parameters the likelihood function (dashed lines) is quite erratic with a plethora of local maxima. This behavior is evident for the parameters ϕ, ψ, π and τ. Obviously, these characteristics entail problems in maximizing the likelihood function using gradient-based optimizers. Looking at the corresponding dimensions of $\hat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}_k)$ (solid lines), we note that the EM algorithm produces smooth and convex functions with maxima close to the true values which are easier found in the optimization process. It is also interesting that the true observed-data log-likelihood function subject to σ_ε^2 seems to be a continuously growing function whereas the corresponding Q-function is convex and contains a global maximum near the true value. The log-likelihood functions subject to ι^2 and κ are rather flat even when using the EM algorithm so that exact estimation of these parameters remains difficult, explaining the poor estimates.

[19] Using the prediction error decomposition, the particle-filtering approximation of the observed-data log-likelihood function is given by $\log L_{\boldsymbol{\theta}_k}(y_{1:T}) = \sum_{t=1}^{T} \log p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}) \approx \sum_{t=1}^{T} \log(\sum_{i=1}^{N} \hat{w}_t^{(i)}/N)$, where $\hat{w}_t^{(i)}$ is defined as in Eq. (3.11) (see Creal, 2012, p. 274).

A matter of major concern are the log-likelihood functions subject to the parameter τ. Besides a rough surface the observed-data log-likelihood function also contains a step shape. This unfavorably shaped graph may explain our identification problem. Even the EM algorithm does not seem to be able to overcome these functional problems and the approximated Q-function remains a step function. Owing to this curvature we are not able to identify τ globally. This identification issue can lead to difficulties when estimating bubbles in real-world data. However, the illustrated approximated Q-function subject to τ is characterized by two large plateaus indicating that we have no significant change in the likelihood of the data for a wide range of τ-values (i.e. for τ ranging between 0 and 16 and 20 and 32). In this case, the likelihood and the other parameter estimates are not sensitive to changes in τ. This fact can be exploited practically by determining a grid of τ-values and running estimation with each fixed τ. Using a goodness-of-fit test, we may then indirectly identify a suitable parameter value for τ.[20]

Figure 3.3 about here

A further objective of this paper consists in estimating the periodically collapsing bubble time series by filtering the unobservable bubble process from stock prices and dividends. Consequently, we are interested in the most precise filtering of the bubble component. Based on the estimated parameter values, Figure 3.3 displays (a) the filtered bubble path using the particle filter (upper panel, solid line), (b) the estimated bubble path using the particle smoother (lower panel, solid line) and (c) the true (simulated) bubble process (both panels, dashed lines). For both estimations we used $N = 500$ particles. Despite the poor parameter estimates of ι^2

[20] We tackle this problem in Section 3.5.4.

and κ, it is obviously possible to estimate the periodically collapsing bubble process accurately. Comparing both methods, we find that the particle smoother maps the true bubble process almost exactly. Evidently, this procedure enables us to estimate the latent bubble process from the data nearly exactly. Based on this estimated bubble time series, we may identify the emergent phase, the bubble's peak values as well as its bursting dates.

3.5.4 Goodness-of-fit and model diagnostics

A major concern in the estimation of the model from Eqs. (3.1) and (3.2) is the non-identifiability of the parameter τ. However, the likelihood function of the model and the estimation results do not appear to be very sensitive to variations in τ. In order to evaluate this impact in more detail, we apply a goodness-of-fit test. Our approach is twofold. First, we use a goodness-of-fit test to indirectly identify a suitable parameter value for τ. Second, we check how well the economic model fits the data in general.

A powerful test to evaluate the goodness-of-fit of a model is provided by Diebold et al. (1998). The test is based on the well-known probability integral transform due to Rosenblatt (1952) and assesses the model fit by evaluating a sequence of one-step-ahead density forecasts. The basic idea may be described as follows. Let $\{y_t\}_{t=1}^T$ be a series of realizations from a sequence of the densities $\{f_t(y_t \mid \Omega_{t-1})\}_{t=1}^T$, conditioned on an information set Ω_{t-1}. If the sequence of forecast densities of an assumed model $\{p_t(y_t \mid \Omega_{t-1})\}_{t=1}^T$ coincides with the true densities of the data generating process,

that is if $\{p_t(y_t \mid \Omega_{t-1})\}_{t=1}^T = \{f_t(y_t \mid \Omega_{t-1})\}_{t=1}^T$, then the sequence of probability integral transforms (PITs)

$$z_t = \int_{-\infty}^{y_t} p_t(u \mid \Omega_{t-1}) du \tag{3.47}$$

should be i.i.d. $U(0,1)$ distributed (see Diebold et al., 1998).

Although originally established for out-of sample evaluations, we use this test in-sample to test for specification adequacy. The advantage of this procedure is that we do not compare our model with any reference model, but directly check how well our model fits a given data set. Any deviation from the i.i.d. $U(0,1)$ distribution may indicate wrong distributional assumptions on the underlying data generating process, poorly captured conditional dynamics or both, whereas a mere rejection of the i.i.d. condition indicates a poorly specified conditional dynamics (see Tay and Wallis, 2000, p. 250). Overall, the method provides information about the deficiencies of distributional assumptions and thus may provide hints for improvement (see Elliott et al., 2006, p. 208).

There are several approaches to checking whether the sequence $\{z_t\}_{t=1}^T$ is i.i.d. $U(0,1)$ scattered in the literature. A visual approach consists in inspecting the cumulative distribution function (cdf) of the PITs which should coincide with the 45-degree line on the interval [0,1]. Additionally, we may use the Kolmogorov-Smirnov (KS) test to measure the deviations from the 45-degree line. However, it is important to note that the null hypothesis underlying the KS test is a joint hypothesis of i.i.d. uniformity and little is known about the impact of departures from the independence assumption (see Tay and Wallis, 2000, p. 250). Hence, a rejection of the KS test may be caused by (a) a violation of uniformity, (b) a violation of the i.i.d. assumption, or (c) both violations (see Diebold et al., 1998,

p. 869).[21] In view of this, we separately check the i.i.d. condition by the use of a Ljung-Box (LB) test with 20 lags. The corresponding critical values at the 1%, 5% and 10% significance levels are 37.57, 31.41 and 28.41, respectively. Overall, we use three different testing methods, one visual and two formal tests, to evaluate the goodness-of-fit.[22]

To implement our goodness-of-fit test we need the one-step-ahead forecast densities of our model. In our nonlinear state-space framework, the forecast densities $p(y_t \mid y_{1:t-1}; \boldsymbol{\theta})$ correspond to the contributions of the observed-data log-likelihood given by the prediction error decomposition (see Creal, 2012). It can be shown that an estimate of the forecast density at time t is a byproduct of the particle filter given by

$$p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}) \approx \sum_{i=1}^{N} \tilde{w}_{t-1}^{(i)} \hat{w}_t^{(i)}. \tag{3.48}$$

In the case of the particle filter using a resampling step in each period, we obtain

$$p(y_t \mid y_{1:t-1}; \boldsymbol{\theta}) \approx \frac{1}{N} \sum_{i=1}^{N} \hat{w}_t^{(i)}, \tag{3.49}$$

where $\hat{w}_t^{(i)}$ is defined as in Eq. (3.11) which coincides with the density of the observation equation from Eq. (3.45). Using the observations of P_t and D_t, the particles $\{x_t^{(i)}\}_{i=1}^N$ from the sequential importance sampling (as estimate of B_t) and the parameter estimates, we can then evaluate the forecast density at each time t.

Table 3.2 about here

[21] We will use a two-sample Kolmogorov-Smirnov test implemented in the software package MATLAB. For the distribution under the null hypothesis we use a sample of one million i.i.d. uniforms. The critical value at a 5% level is given by $1.36\sqrt{(1000000+T)/(1000000 \cdot T)}$.

[22] Diebold et al. (1998) suggest additional graphical methods to identify potential deficiencies of a model so that they are able to calibrate it. For example, they examine the correlograms of higher moments of the PITs to detect particular conditional dependencies.

Next, we apply the test to examine the relevance of the parameter τ for parameter estimation and model fit. Using the simulated data set, we estimate the nonlinear state-space model with different τ-values and evaluate the results with the goodness-of-fit test.[23] Table 3.2 shows the results of the parameter estimation with $\tau = 1, 5, 15, 25, 35$.[24] In line with the analysis of the log-likelihood functions (see Section 3.5.3) we expect similar results of the parameter estimates when $\tau = 1, 5, 15$. The corresponding parameter estimates of ϕ, ψ and π range within a band width of 2% around the true values. By contrast, the estimates of σ_ε^2, κ and ι^2 deviate more from the true values. Nevertheless, all parameter estimates are significant at conventional levels.[25] For $\tau = 25$ and $\tau = 35$ some parameter estimates deviate much more from the true values, especially the estimates of ϕ, σ_ε^2 and κ. Furthermore, for $\tau = 35$ the standard error of the parameter κ appears to be rather high.

Figure 3.4 about here

Next, we use each estimated parameter set with $\tau = 1, 5, 15, 25, 35$ and apply the particle-filter approach to estimating the latent bubble process from the simulated stock prices and dividends. Figure 3.4 shows the filtered bubble processes using the particle smoother with $N = 500$ particles (black lines), as well as the true (simulated) bubble process (blue line). For $\tau = 1, 5, 15$ we obtain good filtering results and the true bubble process is almost perfectly mapped. Using the estimated parameter set with $\tau = 25, 35$, we find that the filtered bubble process is upward shifted. Although the general structure of the true bubble is matched, these two estimated bubble processes are overestimated.

[23] We use 500 particles for each evaluation.

[24] The number of particles, the convergence criterion and the initial parameter vector were chosen as described in Section 3.5.3.

[25] Some of the differences can be ascribed to the use of a Monte Carlo approach (see Schön et al., 2011, p. 47).

Figure 3.5 about here

Table 3.3 about here

Finally, we use the goodness-of-fit test to analyze the impact of τ on the model fit. Figure 3.5 displays the corresponding cdfs of the PITs for the parameter and state estimates for $\tau = 1, 2, 5, 15, 25, 35$. The confidence intervals are computed by the critical value at a 5% significance level of the KS test which equals 0.08589. The estimates for $\tau = 1, 2, 5, 15$ do not exhibit a significant difference between the cdfs of the PITs and the true cdf of a $U(0, 1)$ distribution. The values of the LB test statistics are similar and we cannot reject the null hypothesis of no correlation in the sequence of PITs (see Table 3.3). These results are confirmed by the KS test given in Table 3.3.

For $\tau = 25$ we cannot reject the $U(0, 1)$ condition, but the KS test statistic has increased and the LB test rejects the i.i.d. condition at the 10% level (see Table 3.3). For $\tau = 35$, the cdf of the PITs exhibits large deviations from 45-degree line and the KS test statistic has increased sharply. Thus, we can now reject the i.i.d. $U(0, 1)$ distribution at the 1% level. Furthermore, the LB test rejects the i.i.d. condition at the 1% level. As a result, we find similarly good model fits for $\tau = 1, 2, 5, 15$, which are in line with the results from the log-likelihood analysis. The results for $\tau = 25$ and $\tau = 35$ reveal deficiencies of the estimated model, in particular for $\tau = 35$. For these values of τ we obtain poor parameter and state estimates indicating misspecifications and the model is no longer able to capture the true distribution and dynamics of the underlying data generating process.

Based on these results, we conclude that our parameter and state estimates as well as the goodness-of-fit evaluations are not very sensitive to variations in the

parameter τ. If we choose τ far away from the true parameter, we obtain parameter estimates leading to an overestimated latent bubble process. However, the use of the goodness-of-fit test theoretically enables us to identify these bad choices of τ. This may be an advantage when analyzing real-world data sets, as it is not necessary to have exact knowledge of τ. In practice, we define a very rough grid for τ and use the goodness-of-fit test to identify an appropriate set of parameters providing accurate estimates of the latent bubble process.

3.5.5 Real-world data

In this section we apply the estimation procedure to a real-world data set consisting of real prices and dividends of four major stock-price indices: the German stock index (DAX), the Standard and Poor's index (S&P 500), the National Association of Securities Dealers Automated Quotations index (NASDAQ composite) and the Hang Seng index (HSI). All time series cover the period from January 1981 to February 2014 (398 observations).[26]

Figure 3.6 about here

Following Homm and Breitung (2012) we calculate the dividend time series by multiplying the respective stock-price index by the corresponding dividend yield. The nominal data are transferred to real data by the corresponding consumer-price index. The data for the DAX, NASDAQ composite and HSI are taken from *Datastream*. The data sets of the S&P 500 were compiled from Robert Shiller's website.[27]

[26] In line with our simulation study we assume for our estimation procedure that there is either a very small or even no bubble at the beginning of our observation period. All time series start in 1981 since this phase is characterized by low or moderate price movements.

[27] See http://www.econ.yale.edu/∼shiller/data.htm.

All data sets are provided on a monthly basis. To achieve numerical stabilization of the EM algorithm we divided the time series of the DAX, NASDAQ and HSI by 10 and the time series of the S&P 500 by 20. The data are displayed in Figure 3.6.

Table 3.4 about here

Table 3.5 about here

Table 3.6 about here

Table 3.7 about here

3.5.6 Estimation results of the real-world data set

We begin with the identification of the nonlinear state-space model. We follow our estimation strategy described in the preceding sections and use the goodness-of-fit test provided by Diebold et al. (1998) to indirectly identify an appropriate value of the non-identifiable parameter τ. The Tables 3.4 to 3.7 display the parameter estimates of the four stock-price indices for $\tau = 2, 5, 10, 20$. We initiated the EM algorithm with $\boldsymbol{\theta_0} = (\phi = 20, \sigma_\varepsilon^2 = 0.5, \psi = 0.85, \iota^2 = 0.005, \kappa = 0.5, \pi = 0.8, \tau = 2)$ and computed the standard errors by the stable estimator of the information matrix established by Duan and Fulop (2011).[28]

Tables 3.4 to Table 3.7 show that the parameter τ has a moderate impact on the respective parameter estimates. Overall, we obtain similar index-specific estimates (a) for the DAX when $\tau = 10$ and $\tau = 20$ (Table 3.4), (b) for the NASDAQ when $\tau = 2, 5$ and $\tau = 10, 20$ (Table 3.5), (c) for the S&P 500 for each τ considered (Table

[28]For some estimations the starting value for ψ was chosen as 0.80. Furthermore we used 500 particles for the computation of the standard errors and considered $l = 15$ lags.

3.6) and (d) for the HSI when $\tau = 2, 5$ and $\tau = 10, 20$ (Table 3.7). Irrespective of the τ-value we find significant estimates of the constant fundamental parameter ϕ ranging between approximately 9.9 and 27.1 for all indices. The variances of the fundamental error term σ_ε^2 appear to be low in some cases and range between 0.15 and 2.3. For each index we obtain fairly stable estimates of the parameter ψ. Only for the DAX with $\tau = 2$ and the HSI with $\tau = 2, 5$ the estimates of the parameter π are strictly smaller than 1. Furthermore, only for the HSI with $\tau = 5$ we obtain significant estimates of the parameter κ.

Figure 3.7 about here

Table 3.8 about here

Next, we visually inspect the cdfs of the PITs and use the LB and KS test to indirectly identify appropriate parameter sets. Figure 3.7 displays the particular cdfs of the PITs. The confidence intervals are computed by the critical value at the 5% level using the KS test which equals 0.06808. Obviously, the cdfs deviate from a $U(0, 1)$ distribution for all indices and each value of τ. The results of the KS and LB test are shown in Table 3.8. The KS test rejects the i.i.d. $U(0, 1)$ hypothesis for all indices and all values of τ at a 1% or 5% significance level. The LB test rejects the i.i.d. hypothesis in almost all cases at the 1% or 5% levels. Only for the NASDAQ with $\tau = 5$ the i.i.d. hypothesis cannot be rejected.

The overall results indicate that our suggested economic model does not fit the real-world data very well. Obviously, there are some types of misspecification in the distributional assumptions and model dynamics that cannot only be attributed to the parameter τ.[29] Despite of this poor model fit, we may use the results to

[29] We discuss these misspecifications in Section 3.5.7.

choose an appropriate parameter set for each index to estimate the latent bubble component. For the DAX we use the parameter estimates associated with $\tau = 10$. Based on the visual inspection of the cdfs and the KS test statistics we are not able to identify a superior parameter set. However, for $\tau = 10$ the LB test rejects the i.i.d. hypothesis only at the 5% level. Similarly, for the NASDAQ we choose the estimates associated with $\tau = 5$ since the LB test does not reject the i.i.d. hypothesis at any conventional level. For the S&P 500 we obtain similar cdfs and test statistics for each τ. We choose the parameter set associated with $\tau = 20$ since these parameter values entail the smallest value of the KS test statistics. For the HSI the KS tests indicate slightly favorable results for $\tau = 5, 10, 20$. However, visual inspection on the cdfs suggests a relatively good model fit using the parameter estimates associated with $\tau = 5$.

All chosen sets of parameter estimates appear to indicate the presence of a speculative bubble component since the estimates of the bubble-specific discount factors ψ and the variance parameters ι^2 are significantly different from zero. However, only for the HSI we may conclude that the estimated bubble process follows the Evans model because the parameter π is estimated as $0.99 < 1$. For the DAX, NASDAQ and S&P 500, the probability of a bubble collapse is estimated as zero (since $\hat{\pi} = 1$). Consequently, the first and second phase of the Evans bubble are identical and the bubble's econometric structure reduces to the simpler model[30]

$$B_t = \frac{1}{\psi} B_{t-1} u_t. \tag{3.50}$$

The fact that the reduced bubble process (3.50) does not include any periodically collapsing structure does not necessarily imply that the bubble never collapses. The

[30] In this case the parameter κ is no longer relevant for the model because both states are equal. This is emphasized by our estimation results, since in these cases κ is not significantly estimated.

bubble process (3.50) can adjust stochastically due to the lognormal error term u_t, but never collapses periodically (almost) completely within one month. Large values of the variance parameter ι^2 may lead to large fluctuations in the bubble process. Furthermore, the bubble satisfies the (discounted) martingale property and is thus consistent with a rational bubble.

Figure 3.8 about here

Based on the parameter estimates we now disentangle the latent bubble process from the stock-price indices and dividends by using the particle-filter and the particle-smoother approach. Figure 3.8 displays the respective bubble processes estimated via the particle smoother with $N = 500$ (solid lines) and the corresponding fundamental processes (dashed lines).[31] First, we note that the filtered bubble processes closely follow the dynamics of the stock-price indices. Since variance estimates of the fundamental error term (σ_ε^2) are small (Table 3.4 to 3.7), the fundamental values themselves are slightly more volatile than the dividend processes. Obviously, the largest portion on index fluctuation may be ascribed to the presence of the bubble. For the DAX, NASDAQ and S&P 500 we conclude that at the beginning of the observation period only a small part of the stock prices are affected by rational bubbles. Then, the bubble components turn to become the dominating share of the price indices, especially during the new-economy period (1995-2000) and around the subprime mortgage and credit crisis (2007-2008). Furthermore, several adjustments took place during the observation period, but the bubble components never collapses (almost) completely within one month. This result appear to be realistic in view of historical data on stock prices and the financial-crises literature which indicate

[31] The fundamental series are competed as the difference between the stock prices and the estimated bubble values.

that stock-price adjustments in the aftermath of a crash typically are longer lasting processes (see Allen and Gale, 2000).

Analyzing the HSI we find several more or less pronounced speculative phases within the observation period. In contrast to the other indices, the bubble process collapses periodically (almost) completely which corresponds to the dynamics of the estimated Evans specification (Table 3.7). These collapses are particularly obvious during the periods of the Asian crisis (1997-1998) and the subprime mortgage crisis (2007-2008).

Figure 3.9 about here

Following the lines of Wu (1997) and Brooks and Katsaris (2005), we calculate the ratios of the filtered bubble components and the stock-price indices. Since we estimated our bubble process by means of the the standard present-value model, we interpret this ratio as the speculation share in the stock markets considered. Figure 3.9 displays the ratios for our four stock-price indices. Obviously, the stock-price bubbles persistently account for substantial parts of the DAX, NASDAQ and S&P 500. Even at the beginning of the observation period approximately 40% of the DAX and NASDAQ and 14% of the S&P 500 may be attributed to a speculative bubble. The DAX ratio peaked in February 2002 (88%) and the S&P 500 in August 2000 (84%). For the NASDAQ the ratio attains its maximum in September 2000 with a value of 97% indicating that the speculative component dominated the stock index almost completely. The highest decreases in the speculation share occur during the last major crisis in 2007-2008 with -28% for DAX, -15% for the NASDAQ and -19% for the S&P 500. Irrespective of the adjustments, these ratios have persisted on a high level (on average 74% to over 82% since the late 1990s). At the end of

the observation period in February 2014 the bubble still accounted for 71% to 76% of the indices.

By contrast, the HSI proportion of the speculative bubble appears comparably low with a peak at 64% in September 2000. However, the HSI ratio shows several major adjustments during the observation period. Evidently, a bubble burst seem to trigger almost complete market corrections, but there still remains a speculative portion between 6% to 21%. Since the beginning of 2013 the bubble has accounted for 34% of the HSI on average.

3.5.7 Model critique

The diagnostic tests based on the goodness-of-fit techniques presented in the preceding section reveal a rather unsatisfactory fit of our nonlinear state-space model to real-world stock-price indices. Typically, the tests lead to the rejection of the i.i.d. assumption and also appear to indicate misspecified distributional assumptions. The crucial question now is why we obtain these results.

A simple explanation may be that real-world stock-price indices cannot be represented by a simply structured fundamental value consisting of a constant multiplied by the current level of dividends. Although further model assumptions might be questioned, we now address an alternative explanation, namely that of a potentially misspecified bubble component.

Our empirical analysis of the real-world stock-price indices from above clearly indicates that real-world bubbles typically deflate rather slowly during several periods and not as abruptly as predicted by the Evans specification. Although the Evans bubble constitutes a realistic and valuable model for describing periodically

collapsing bubbles, it does not provide any room for moderately or strongly delayed bubble deflation.

Finally, we summarize that our nonlinear state-space framework fails to capture several features of the underlying data-generating process (see Tay and Wallis, 2000, p. 250). This, however, does not mean that our framework (a) is misspecified, (b) lacks explanatory power, or (c) is better or worse than other existing models (see Wu, 1997; Brooks and Katsaris, 2005; Al-Anaswah and Wilfling, 2011). In fact, our model is consistent with asset-pricing theory and also with the theory on rational bubbles. It is precisely this latter aspect which makes our framework benefit from an explicit economic model structure that enables us to interpret the estimated bubble process as well as the bubble parameters unambiguously.

3.6 Conclusion

In this chapter we estimate the econometric structure of periodically collapsing stock-price bubbles and filter the latent bubble component from stock-prices and dividends. To this end, we incorporate the Evans' (1991) bubble specification into the standard present-value model. We transform this framework into a nonlinear state-space model which is a valid solution to the Euler equation.

To estimate this nonlinear state-space model we use particle-filter methods. Since we avoid the usual log-linearization of the model, our estimates provide a clear-cut relationship between the stock price, its fundamental value and the bubble component that follows directly from the standard present-value model. As a result, the estimated bubble process with all its parameters can be interpreted unambiguously in line with economic theory.

Using artificial data, we demonstrate that our estimation framework is capable of (econometrically) identifying the underlying nonlinear state-space specification. If the stock price is driven by a periodically collapsing bubble, this unobservable bubble process is precisely estimated from the simulated data. Due to the economic structure of the underlying present-value model, we are able to identify the emergent phase, the peak values of bubble as well as the dates of bursting.

Finally, we apply our econometric procedure to a real-world data set consisting of real stock-prices and dividends for the DAX, the S&P 500, the NASDAQ and the HIS. For all indices our parameter estimates indicate the presence of rational speculative bubbles. However, we only find evidence in favor of an explicit Evans (1991) bubble for the HSI.

The diagnostic tests of our estimated model for the real-world data set appear to reveal some misspecifications of our framework. We argue that these problems are due to the specification of the Evans bubble which does not provide any room for slowly or moderately deflating bubbles. As a result, we suggest improving the goodness-of-fit by using an alternative rational bubble specification which is able to capture slowly deflating bubble processes. In Chapter 5 we propose an explicit bubble specification that is capable of overcoming these deficiencies.

Figures

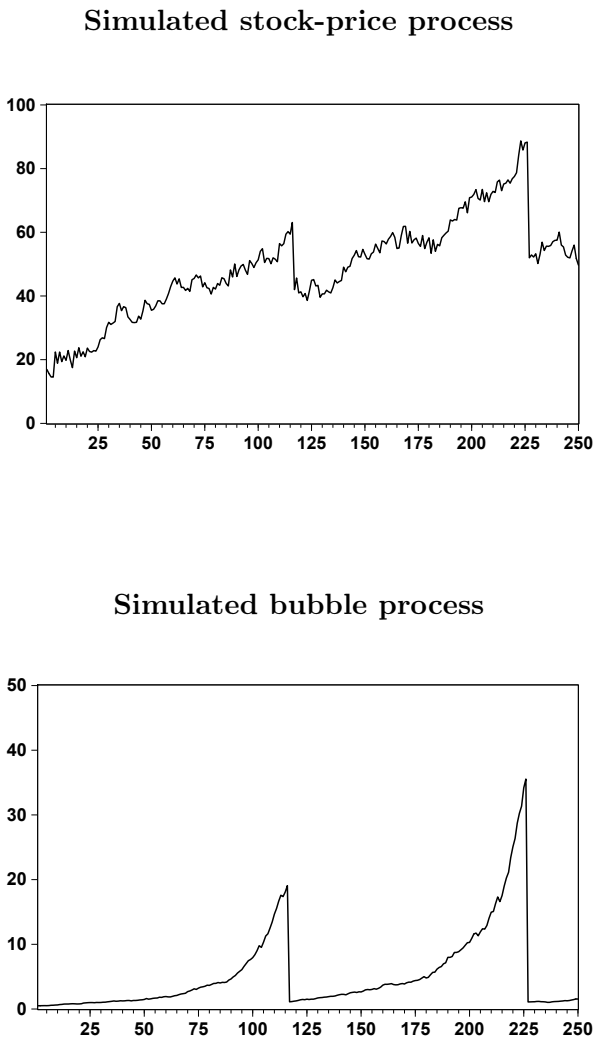

Figure 3.1: Stock-price process and included Evans bubble.

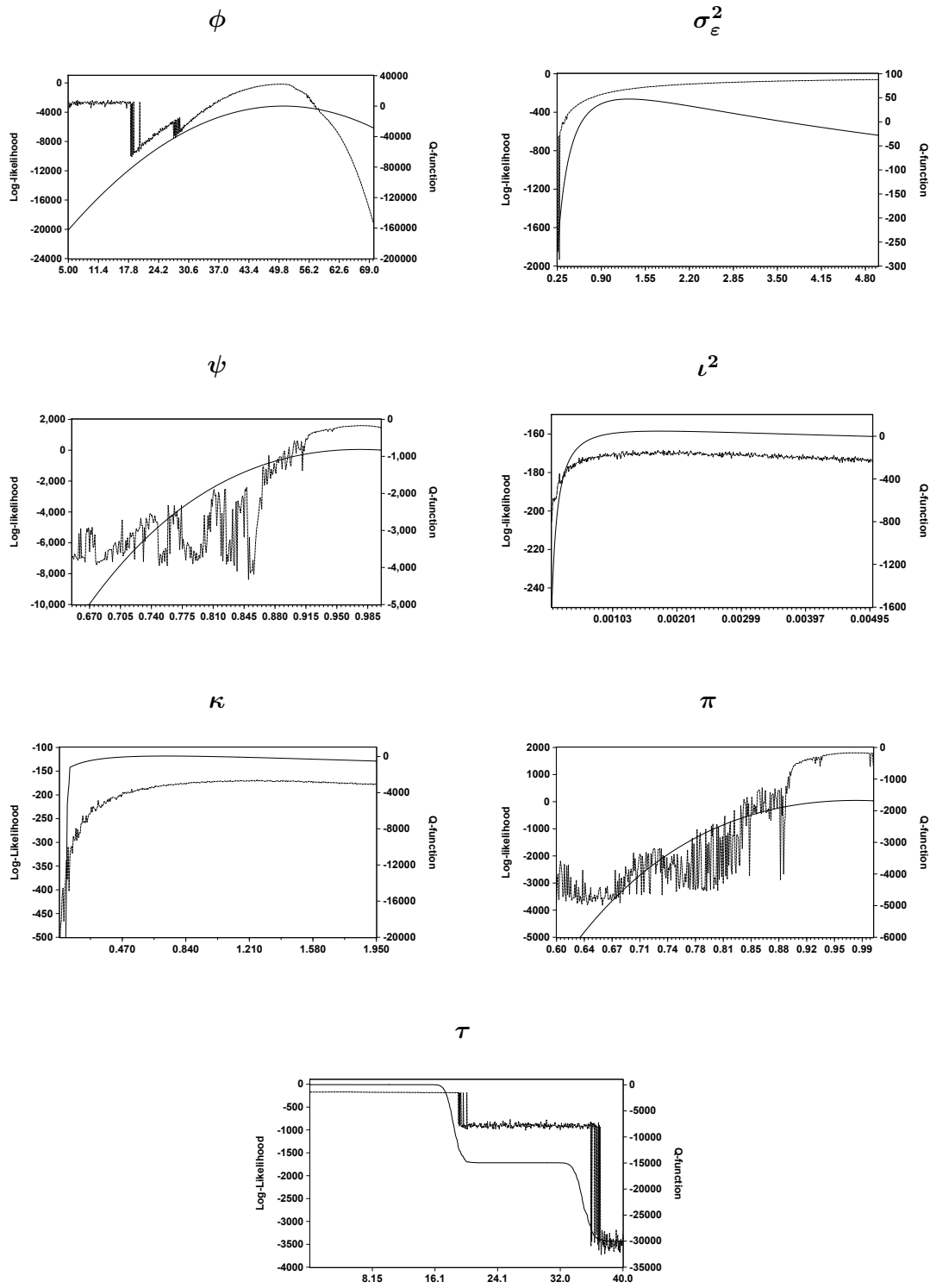

Figure 3.2: True log-likelihood (dashed lines) and approximated Q-function (solid lines) as functions of the parameters.

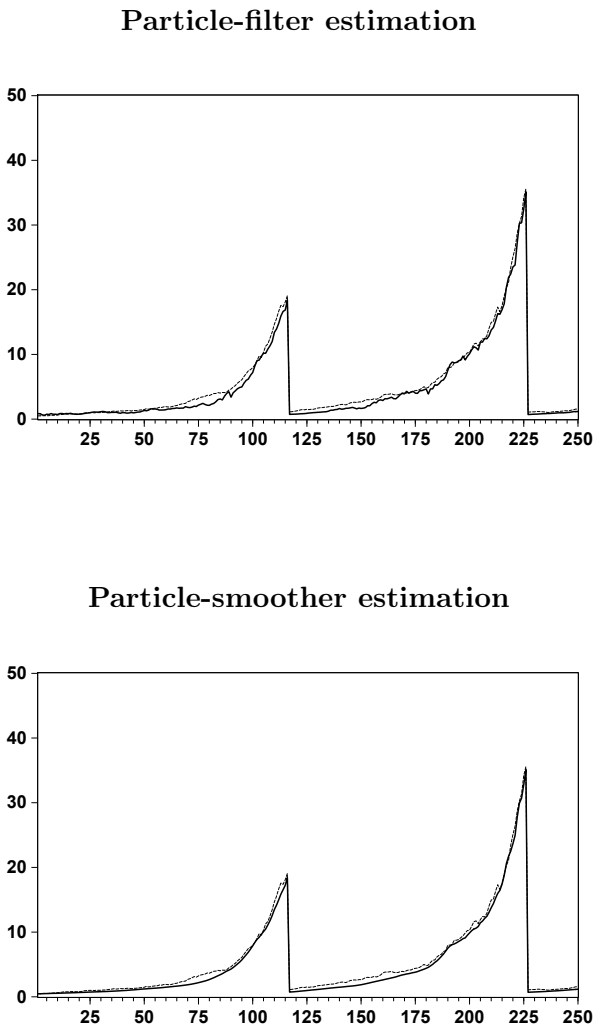

Figure 3.3: Estimated Evans-bubble process (solid lines) versus true Evans-bubble process (dashed lines).

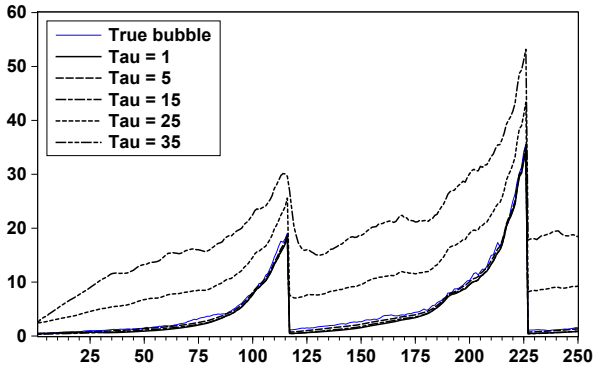

Figure 3.4: Estimated Evans-bubble processes for $\tau = 1, 5, 15, 25, 35$ (black lines) versus true Evans-bubble process (blue line).

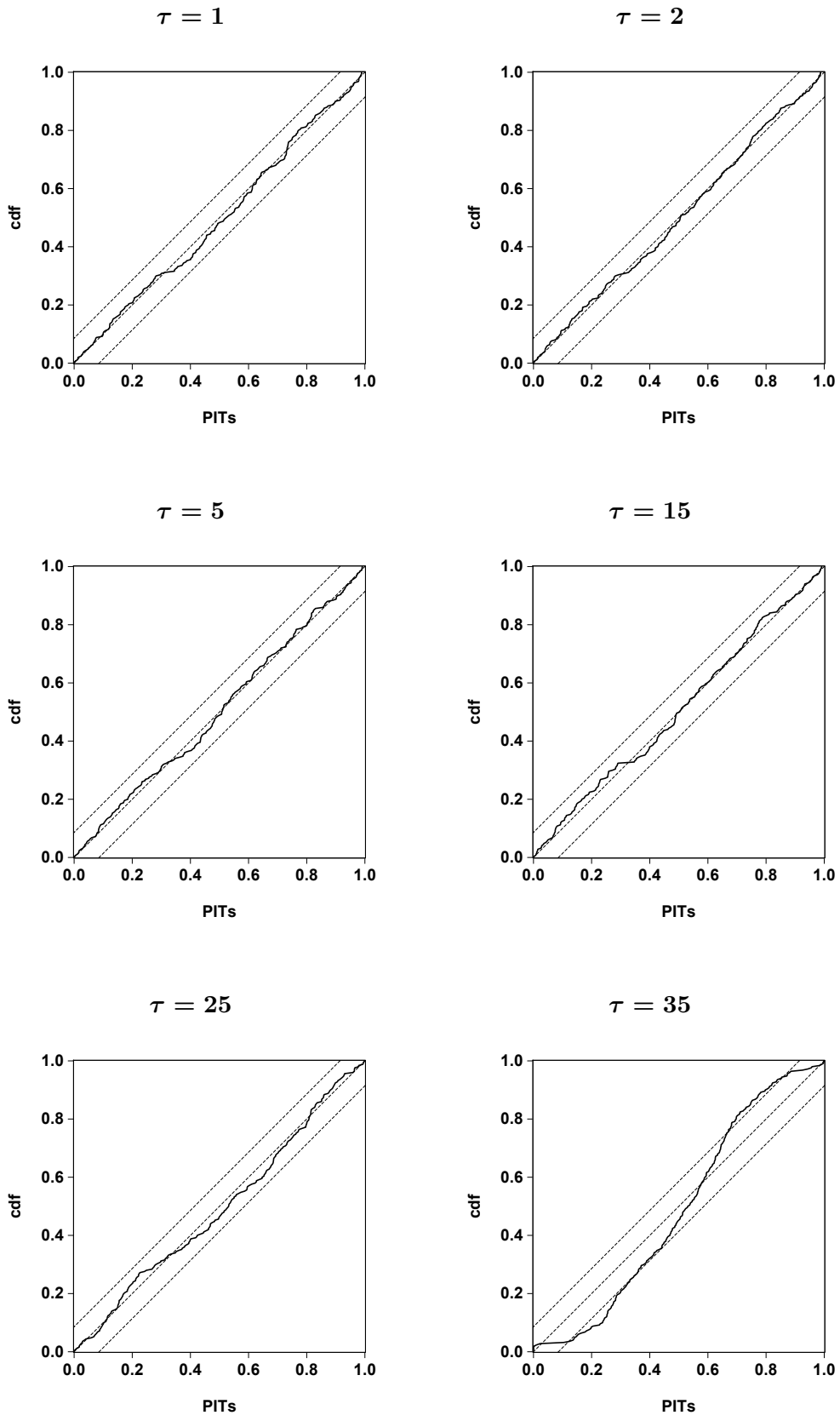

Figure 3.5: Cdfs of the PITs for the estimated parameters for $\tau = 1, 2, 5, 15, 25, 35$. True cdfs and 5% confidence intervals are represented by dashed lines.

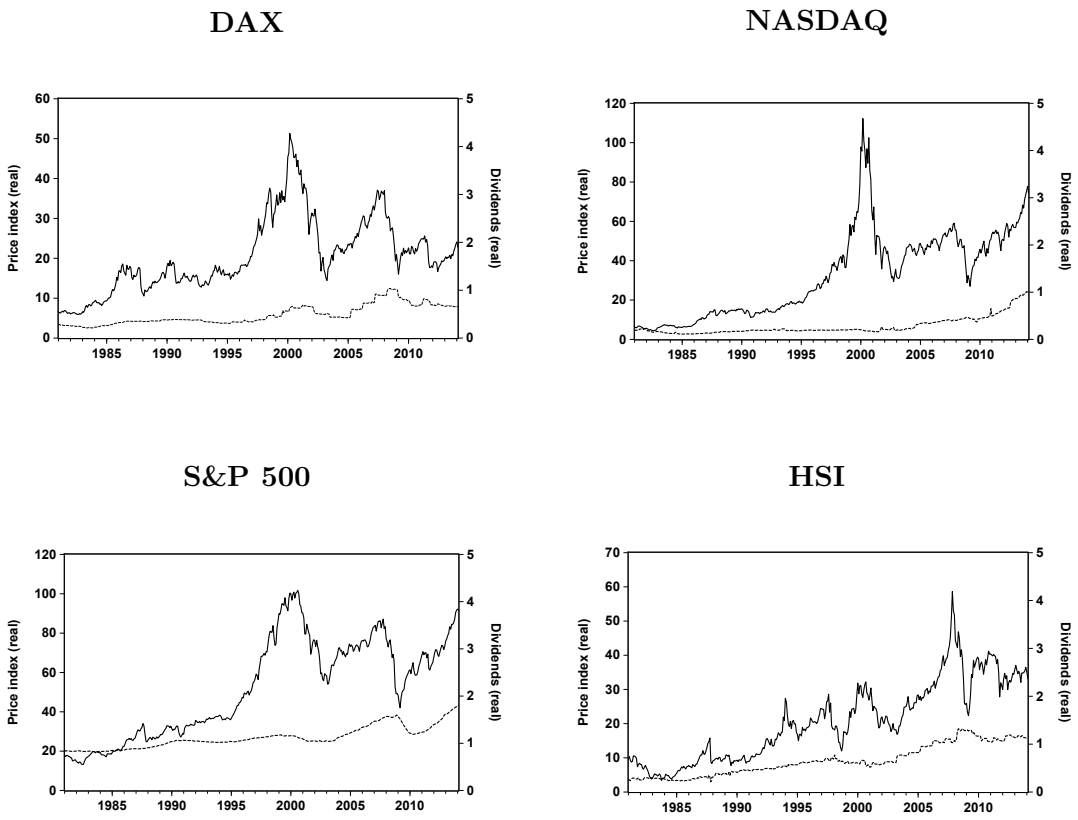

Figure 3.6: Price indices (solid lines) and dividends (dashed lines) of the DAX, NASDAQ, S&P 500 and HSI.

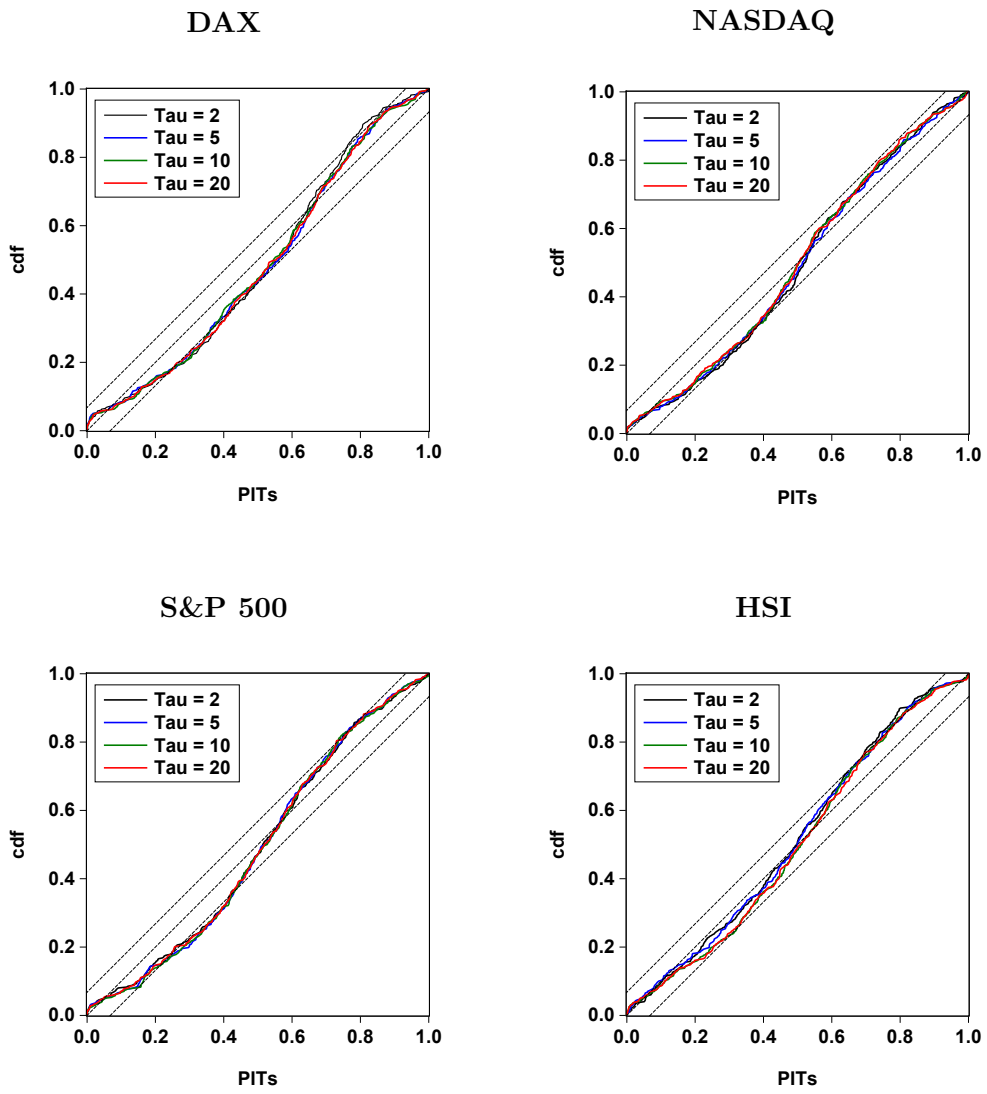

Figure 3.7: Cdfs of the PITs for the DAX, NASDAQ, S&P 500 and HSI ($\tau = 2, 5, 10, 20$). True cdfs and 5% confidence intervals are represented by dashed lines.

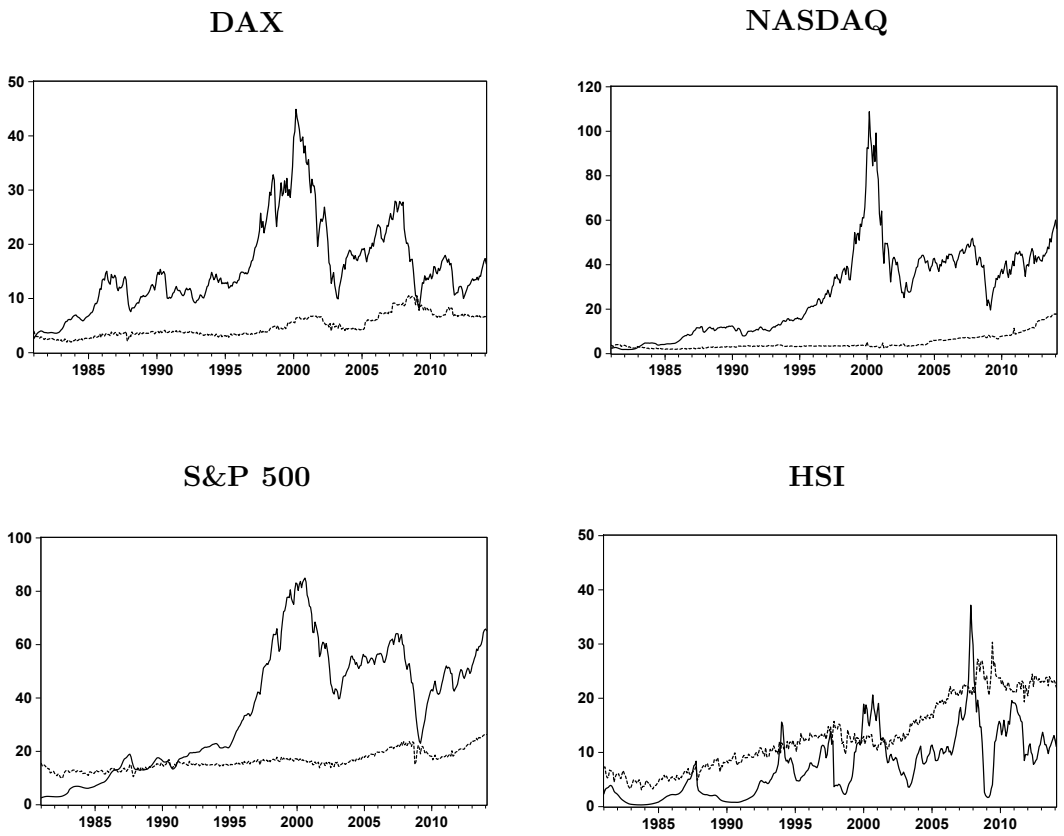

Figure 3.8: Estimated Evans-bubble processes (solid lines) and the fundamental processes (dashed lines) for the DAX, NASDAQ, S&P 500 and HSI.

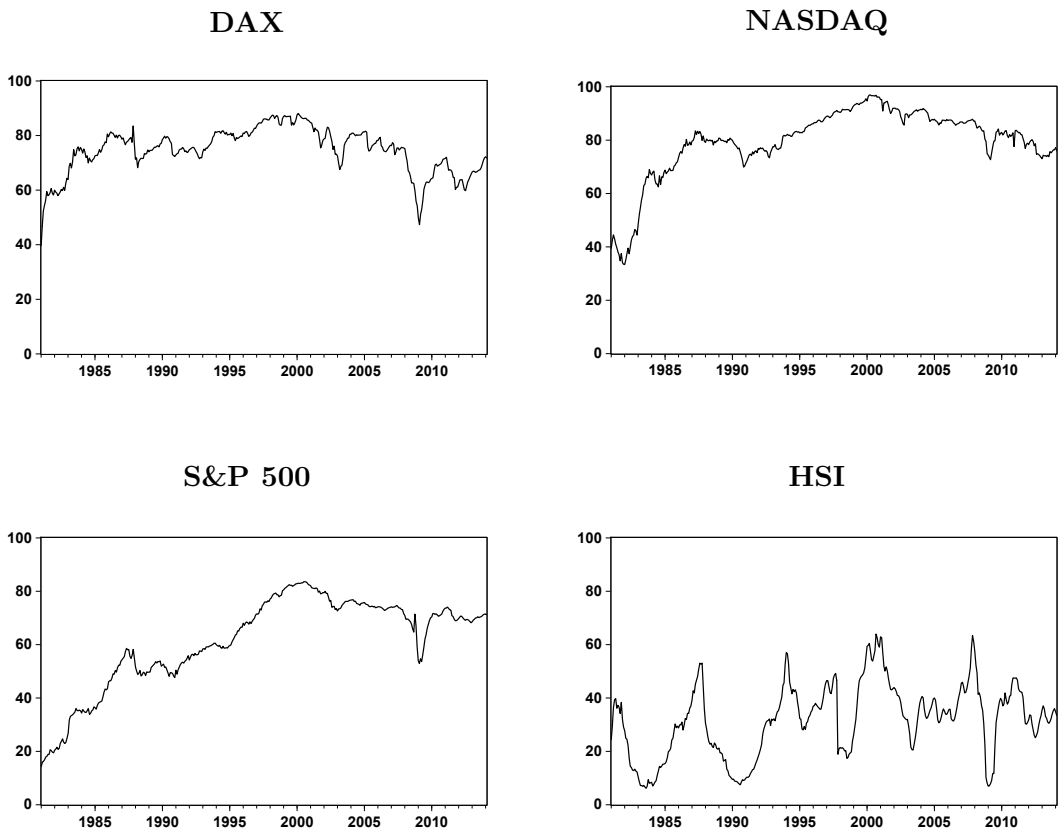

Figure 3.9: Ratios of the Evans-bubble processes and the stock-price series for the DAX, NASDAQ, S&P 500 and HSI.

Tables

Parameter	True value	Estimate	Standard error
ϕ	50.0000	50.6970	0.0146
σ_ε^2	1.2000	1.3008	0.0249
ψ	0.9804	0.9762	7.4661×10^{-5}
ι^2	0.0010	0.0018	3.2522×10^{-5}
κ	1.1000	0.7139	0.0048
π	0.9800	0.9769	8.9772×10^{-5}

Table 3.1: Parameter estimates (for $\tau = 2$) using the EM algorithm.

Paramter	$\tau = 1$	$\tau = 5$	$\tau = 15$	$\tau = 25$	$\tau = 35$
ϕ	50.9803	50.4976	50.9018	43.0652	33.7826
	(0.0143)	(0.0157)	(0.0162)	(0.0448)	(0.2375)
σ_ε^2	1.3144	1.2923	1.2887	1.3404	4.4719
	(0.0240)	(0.0253)	(0.0223)	(0.0213)	(0.1248)
ψ	0.9724	0.9720	0.9643	0.9837	0.9867
	(9.4995×10^{-5})	(3.4945×10^{-5})	(1.9920×10^{-5})	(6.6439×10^{-6})	(3.4647×10^{-5})
ι^2	0.0021	0.0016	0.0027	0.0013	0.0027
	(1.6467×10^{-4})	(1.2424×10^{-5})	(4.3988×10^{-6})	(4.4010×10^{-7})	(5.6556×10^{-7})
κ	0.4661	0.7152	0.4106	8.1593	18.1024
	(0.0075)	(0.0421)	(0.0023)	(0.1093)	(21.9380)
π	0.9812	0.9773	0.9663	0.9555	0.9630
	(1.1812×10^{-4})	(6.2191×10^{-5})	(8.4454×10^{-4})	(0.0014)	(0.0032)
Iterations (k)	67	282	500	286	464

Table 3.2: Parameter estimates using the EM algorithm after k iterations for $\tau = 1, 5, 15, 25, 35$. Standard errors are in parentheses.

Test statistic	$\tau = 1$	$\tau = 2$	$\tau = 5$	$\tau = 15$	$\tau = 25$	$\tau = 35$
KS test	0.0487	0.0348	0.0417	0.0374	0.0541	0.1422***
LB test	24.5677	24.9075	24.6885	23.6554	29.7712*	169.3196***

Table 3.3: KS tests and LB tests on the PITs using the parameter estimates for $\tau = 1, 2, 5, 10, 15$. ***, **, * denote statistical significance at 1%, 5% and 10% levels, respectively.

Parameter	$\tau = 2$	$\tau = 5$	$\tau = 10$	$\tau = 20$
ϕ	19.3068	13.4400	9.9914	9.8929
	(0.0142)	(0.0422)	(0.0173)	(0.0259)
σ_ϵ^2	0.4693	0.2291	0.1815	0.2832
	(0.0011)	(0.0012)	(5.9057×10^{-4})	(0.0013)
ψ	0.9862	0.9914	0.9918	0.9921
	(9.9975×10^{-5})	(1.8229×10^{-4})	(6.3935×10^{-5})	(2.4375×10^{-5})
ι^2	0.0115	0.0075	0.0063	0.0056
	(8.7392×10^{-7})	(3.1709×10^{-7})	(2.1498×10^{-7})	(2.7116×10^{-7})
κ	1.5173	0.4982	0.5829	0.7995
	(12.9632)	$(1.2119 \times 10^{+10})$	$(3.9155 \times 10^{+10})$	$(3.0148 \times 10^{+11})$
π	0.9981	1.0000	1.0000	1.0000
	(1.1593×10^{-4})	(1.2726×10^{-4})	(8.4917×10^{-5})	(3.3025×10^{-4})
Iterations (k)	500	176	360	263

Table 3.4: Parameter estimation results after k iterations for the DAX ($\tau = 2, 5, 10, 20$). Standard errors are in parentheses.

Parameter	$\tau = 2$	$\tau = 5$	$\tau = 10$	$\tau = 20$
ϕ	17.3582	17.6351	27.0851	26.8630
	(0.2688)	(0.0315)	(0.0529)	(0.0453)
σ_ϵ^2	0.4595	0.1508	0.4095	0.2743
	(0.0011)	(1.5869×10^{-4})	(0.0020)	(0.0013)
ψ	0.9888	0.9888	0.9854	0.9855
	(2.5314×10^{-4})	(4.4781×10^{-5})	(2.8326×10^{-5})	(2.4927×10^{-5})
ι^2	0.0054	0.0060	0.0072	0.0077
	(8.5121×10^{-8})	(1.2074×10^{-7})	(1.6427×10^{-7})	(1.3044×10^{-7})
κ	0.6361	0.6428	0.3878	0.6231
	$(1.2613 \times 10^{+9})$	$(8.6702 \times 10^{+9})$	$(4.2222 \times 10^{+10})$	$(2.0991 \times 10^{+11})$
π	1.0000	1.0000	1.0000	1.0000
	(2.0679×10^{-4})	(5.6630×10^{-5})	(6.5553×10^{-5})	(1.0641×10^{-4})
Iterations (k)	166	111	192	320

Table 3.5: Parameter estimation results after k iterations for the NASDAQ ($\tau = 2, 5, 10, 20$). Standard errors are in parentheses.

Parameter	$\tau = 2$	$\tau = 5$	$\tau = 10$	$\tau = 20$
ϕ	14.6754	13.7861	13.4792	14.6202
	(0.0075)	(0.0046)	(0.0061)	(0.0055)
σ_ϵ^2	1.3323	1.7365	1.6102	1.5623
	(0.0045)	(0.0058)	(0.0050)	(0.0047)
ψ	0.9900	0.9901	0.9900	0.9901
	(2.8120×10^{-4})	(6.9530×10^{-5})	(2.2329×10^{-5})	(1.0070×10^{-5})
ι^2	0.0028	0.0025	0.0025	0.0027
	(5.1199×10^{-8})	(4.2373×10^{-8})	(3.7732×10^{-8})	(3.0705×10^{-8})
κ	0.5499	0.5988	0.6047	0.6244
	$(1.4800 \times 10^{+9})$	$(5.3745 \times 10^{+9})$	$(1.2043 \times 10^{+10})$	$(7.7096 \times 10^{+10})$
π	1.0000	1.0000	1.0000	1.0000
	(2.3889×10^{-4})	(5.3357×10^{-5})	(2.7304×10^{-5})	(4.2908×10^{-5})
Iterations (k)	218	358	372	181

Table 3.6: Parameter estimation results after k iterations for the S&P 500 ($\tau = 2, 5, 10, 20$). Standard errors are in parentheses.

Parameter	$\tau = 2$	$\tau = 5$	$\tau = 10$	$\tau = 20$
ϕ	21.7688	20.0985	18.5438	18.0396
	(0.0130)	(0.0159)	(0.0190)	(0.0176)
σ_ϵ^2	2.3405	1.5149	1.0550	1.0005
	(0.0080)	(0.0077)	(0.0059)	(0.0042)
ψ	0.9690	0.9755	0.9792	0.9801
	(9.4410×10^{-5})	(5.7720×10^{-5})	(7.8915×10^{-5})	(7.6289×10^{-5})
ι^2	0.0436	0.0309	0.0329	0.0310
	(2.0208×10^{-5})	(1.0121×10^{-5})	(9.0992×10^{-6})	(6.8028×10^{-6})
κ	0.5435	2.9932	0.5321	0.6460
	(0.5141)	(0.3897)	$(2.5554 \times 10^{+11})$	$(8.8493 \times 10^{+12})$
π	0.9958	0.9900	1.0000	1.0000
	(3.9260×10^{-5})	(6.4080×10^{-5})	(5.0347×10^{-4})	(0.0075)
Iterations (k)	500	500	90	84

Table 3.7: Parameter estimation results after k iterations for the HSI ($\tau = 2, 5, 10, 20$). Standard errors are in parentheses.

	$\tau = 2$	$\tau = 5$	$\tau = 10$	$\tau = 20$
DAX				
KS test	0.0962***	0.0845***	0.0938***	0.0871***
LB test	89.9752***	44.0453***	37.1545**	48.4496***
NASDAQ				
KS test	0.0838***	0.0717**	0.0773**	0.0757**
LB test	40.6307***	26.8094	57.4513***	49.2113***
S&P 500				
KS test	0.0944***	0.1009***	0.0983***	0.0942***
LB test	232.4670***	216.8022***	207.6334***	220.2103***
HSI				
KS test	0.0997***	0.0790**	0.0781**	0.0743**
LB test	305.0413***	185.3675***	130.8157***	123.5147***

Table 3.8: KS tests and LB tests on the PITs of the DAX, NASDAQ, S&P 500 and HSI using the parameter estimates ($\tau = 2, 5, 10, 20$). ***, **, * denote statistical significance at 1%, 5% and 10% levels, respectively.

Chapter 4

Periodically collapsing Evans bubbles and stock-price volatility[1]

4.1 Introduction

Several authors argue that the frequently observed excessive volatility in stock prices may be attributed to the presence of speculative bubbles. Blanchard and Watson (1982) and Flood and Hodrick (1986), *inter alia*, demonstrate in a theoretical framework that bubble components potentially generate excessive volatility. Besides the many articles discussing theoretical aspects of speculative bubbles and econometric techniques for their detection, there is a second strand of literature linking financial crises and/or bubbly periods to stock-price volatility. Two important pieces of research are Brunnermeier and Oehmke (2013) and the so-called Minsky model (as described in Kindleberger and Aliber, 2005, pp. 24-37) according to which financial crises and/or bubbly periods are characterized by different phases of stock-price volatility. During the early stages of a bubbly period stock-price volatility appears

[1]This Chapter is an extended pre-print version of an article published in the Journal *Economics Letters, 123(3), 383-386* (see Rotermann and Wilfling, 2014).

to be low whereas toward the end of the bubble and its burst stock-price volatility is typically high.

In this chapter we consider the existence of periodically collapsing bubbles as proposed by Evans (1991) in the well-known present-value model and theoretically analyze conditional stock-price volatility within this framework. Using the sequential Bayesian Monte Carlo methods, we fit our theoretical model equations to an artificial dataset to gain further insights into stock-price volatility dynamics during bubbly periods. Our analysis has two major findings. First, we show that our rational bubble specification entails excess stock-price volatility. Second, we find that the dynamic structure of this volatility dynamics accords with the phases of low and high volatility as proposed by Brunnermeier and Oehmke (2013) and the Minsky model.

4.2 Conditional stock-price volatility

We consider the linear present-value model as described in Chapter 2.1. From Eq. (2.2) and Eq. (2.5) the stock price at date t is given by

$$P_t = \sum_{i=1}^{\infty} \left(\frac{1}{1+r} \right)^i \cdot E_t(D_{t+i}) + B_t. \tag{4.1}$$

To analyze the impact of rational bubbles on stock-price volatility, we follow Evans (1991) who suggests an empirically plausible class of bubbles that are non-linear, positive, periodically collapsing and satisfy the martingale property (2.6). Recalling the econometric specification of the Evans bubble from Section 3.2, we therefore set

$$B_t = \begin{cases} \frac{1}{\psi} B_{t-1} u_t & , \text{ if } B_{t-1} \leq \tau \\ [\kappa + \frac{1}{\pi\psi}\Big(B_{t-1} - \kappa\psi\Big)\nu_t] u_t \ , & \text{ if } B_{t-1} > \tau \end{cases} . \tag{4.2}$$

In order to compute the conditional volatility associated with the stock-price dynamics given in Eqs. (4.1) and (4.2), it remains to specify a stochastic process $\{D_t\}$ governing the dividend payments. In line with recent literature we assume that dividends follow a driftless random walk of the form

$$D_t = D_{t-1} + \epsilon_t, \tag{4.3}$$

where ϵ_t is an i.i.d. Gaussian white-noise process with mean zero and variance σ^2 (see Al-Anaswah and Wilfling, 2011). Inserting this into Eq. (4.1), we obtain

$$P_t = \beta D_t + B_t = \beta D_{t-1} + B_t + \beta\epsilon_t, \tag{4.4}$$

where $\beta = 1/r$.

We now compute the variance of the stock price P_t given in Eq. (4.4) conditional on all information available to market participants as of date $t-1$, which we denote by $\text{Var}_{t-1}(P_t)$. The associated information set Ω_{t-1} contains, *inter alia*, all past dividends and stock prices. Additionally, we assume that we can observe (or at least estimate) past values of the bubble component.[2] From Eq. (4.4) we have

$$\text{Var}_{t-1}(P_t) = \text{Var}_{t-1}(\beta D_{t-1} + B_t + \beta\epsilon_t) = \text{Var}_{t-1}(B_t + \beta\epsilon_t). \tag{4.5}$$

[2]We will tackle this issue more concretely in Section 4.3.

Since the dividend error process $\{\epsilon_t\}$ is by definition uncorrelated with the bubble process $\{B_t\}$, Eq. (4.5) reduces to

$$\mathrm{Var}_{t-1}(P_t) = \mathrm{Var}_{t-1}(B_t) + \beta^2\sigma^2. \tag{4.6}$$

Obviously, when dividends follow a random walk, conditional stock-price volatility is (up to a constant) completely characterized in terms of the conditional variance of the bubble term. The conditional variance $\mathrm{Var}_{t-1}(B_t)$ itself can be derived from the distributional assumptions of the Evans-bubble specification, see Eq. (3.46). More precisely, the lognormal distribution of u_t from Eq. (4.2) implies that for $B_{t-1} \leq \tau$ the conditional variance of the bubble is given by

$$\mathrm{Var}_{t-1}(B_t) = \left(\frac{1}{\psi}B_{t-1}\right)^2 \cdot \left[\exp(\iota^2) - 1\right]. \tag{4.7}$$

The case $B_{t-1} > \tau$ is slightly more laborious because it involves the two random variables u_t and ν_t. In a first step, Eq. (4.2) allows us to write

$$\mathrm{Var}_{t-1}(B_t) = \mathrm{Var}_{t-1}\left(\kappa u_t + \left[\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right]\nu_t u_t\right). \tag{4.8}$$

Next, we have to take account of the covariance of the variables u_t and $\nu_t u_t$ in Eq. (4.8), which is given by $\pi \cdot (\exp\{\iota^2\} - 1)$. In similar vein, it is straightforward to find $\mathrm{Var}_{t-1}(u_t)$ and $\mathrm{Var}_{t-1}(\nu_t u_t)$. Overall, we obtain for $B_{t-1} > \tau$

$$\begin{aligned}
\mathrm{Var}_{t-1}(B_t) &= \kappa^2 \cdot \mathrm{Var}_{t-1}(u_t) + \left(\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right)^2 \cdot \mathrm{Var}_{t-1}(\nu_t u_t) \\
&\quad + 2\kappa\left(\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right) \cdot \left[\pi \cdot \left(\exp\left\{\iota^2\right\} - 1\right)\right] \\
&= \kappa^2\left(\exp\{\iota^2\} - 1\right) + \left(\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right)^2 \cdot \left(\exp\{\iota^2\} \cdot \pi - \pi^2\right) \\
&\quad + 2\kappa\left(\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right) \cdot \left[\pi \cdot \left(\exp\left\{\iota^2\right\} - 1\right)\right].
\end{aligned} \tag{4.9}$$

Now, inserting Eq. (4.9) into Eq. (4.6), the conditional stock-price variance is given by

$$\mathrm{Var}_{t-1}(P_t) = \begin{cases} \left(\frac{1}{\psi}B_{t-1}\right)^2 (\exp\{\iota^2\} - 1) + \beta^2\sigma^2, & \text{if } B_{t-1} \leq \tau \\ \left[\kappa^2 + 2\kappa\left(\frac{B_{t-1} - \kappa\psi}{\psi}\right)\right] \cdot (\exp\{\iota^2\} - 1) & \\ + \left(\frac{B_{t-1} - \kappa\psi}{\pi\psi}\right)^2 [\exp\{\iota^2\}\pi - \pi^2] + \beta^2\sigma^2, & \text{if } B_{t-1} > \tau \end{cases} . \quad (4.10)$$

4.3 Bubble and stock-price volatility

4.3.1 Theoretical results

To state a first theoretical result we note from Eq. (4.10) that the bubble term B_{t-1} has an increasing effect on conditional stock-price volatility. This implies that the mere existence of a speculative bubble necessarily increases stock-price volatility. We may analyze this impact further by considering the derivative

$$\frac{\partial \mathrm{Var}_{t-1}(P_t)}{\partial B_{t-1}} = \begin{cases} \frac{2}{\psi^2} B_{t-1} \cdot (\exp\{\iota^2\} - 1)\,, & \text{if } B_{t-1} \leq \tau \\ \frac{2}{\psi^2}\left[(B_{t-1} - \kappa\psi) \cdot \left(\frac{\exp\{\iota^2\}}{\pi} - 1\right) + \kappa\psi(\exp\{\iota^2\} - 1)\right], & \text{if } B_{t-1} > \tau \end{cases} . \quad (4.11)$$

Eq. (4.11) establishes a strictly positive relationship between the infinitesimal change in stock-price volatility and B_{t-1}, the bubble level from the previous period. Consequently, an explosive bubble path necessarily entails an explosive path of stock-price volatility.

Next, we address the impact of a bubble burst on stock-price volatility. Since the conditional stock-price variance given in Eq. (4.10) is a function of B_{t-1}, stock-price volatility collapses one period after the bubble burst. Furthermore, owing to Eqs. (4.10) and (4.11) the stock-price volatility process attains its maximum when

B_{t-1} takes on its largest value which typically occurs on the eve of the bubble crash. This theoretical result is consistent with the volatility dynamics described by Brunnermeier and Oehmke (2013) and the Minsky model.

Figure 4.1 about here

4.3.2 Empirical application

Figure 4.1 gives an example of an Evans-bubble process and the stock-price process from Eqs. (4.2) and (4.4) with one large and one moderate bubble. The dividends from Eq. (4.3) follow a random walk with standard deviation $\sigma^2 = 0.0009$. We set the parameter value $r = 0.02$ so that $\beta = 1/r = 50$ while the parameters relevant to the simulation of the Evans bubble are chosen as $\psi = 1/(1+r) = 0.9804$, $\iota^2 = 0.001$, $\kappa = 1.1$, $\pi = 0.98$ and $\tau = 2$.[3] We initiate the bubble process with the value $B_0 = 0.5$. Our artificial dataset consists of 250 observations corresponding to a time span of approximately 21 years on the basis of monthly data.

An important stipulation inherent in our theoretical framework concerns the structure of the information set Ω_{t-1}. We explicitly assume that Ω_{t-1} contains the bubble time series, which is crucial to analyzing stock-price volatility. However, in practice bubble values are unobservable so that we are forced to estimate the bubble process from the data.

To this end, we use the sequential Monte Carlo methods as introduced in Chapter 3.3. This Bayesian approach enables us to estimate a latent variable (our bubble process) from nonlinear and non-Gaussian state-space models. In a first step, similar

[3]The parameter values of the Evans process are different from those given in the original paper by Evans (1991). We choose this parameterization in order to generate only a few big bubbles in the data. In line with the history of financial markets this appears to be a realistic assumption for the length of our data (monthly observations).

to Section 3.5.1, we transform Eqs. (4.2) and (4.4) into a nonlinear state-space form, where Eq. (4.4) describes the observation equation and Eq. (4.2) equals the state equation. The corresponding distribution of the observation equation is Gaussian, with expectation $\beta D_{t-1} + B_t$ and variance $\beta^2 \cdot \sigma_\varepsilon^2$, and the distribution of the state equation is equal to Eq. (3.46) (see Section 3.5.1). Since $\psi = 1/(1+r)$ and $\beta = 1/r$ the parameter vector of this model is given by $\boldsymbol{\theta} = (r, \sigma_\varepsilon^2, \iota^2, \kappa, \pi, \tau)$.

From the nonlinear state-space representation we estimate—besides all other model parameters—the unobservable bubble process. For parameter estimation we apply the EM algorithm as proposed by Schön et al. (2011) (see Section 3.4). We initiated the EM algorithm with a parameter vector $\boldsymbol{\theta_0} = (r = 0.03, \sigma_\varepsilon^2 = 0.5, \iota^2 = 0.005, \kappa = 0.5, \pi = 0.8, \tau = 2)$ and we numerically maximized the Q-function from Eqs. (3.30) and (3.31) by the use of the FMINCON module of the software package MATLAB. For the convergence criterion we used $c = 0.0033$. Additionally, we used the stable estimator of the information matrix established by Duan and Fulop (2011) to compute standard errors as described in Section 3.4.4.[4]

Table 4.1 about here

Table 4.1 displays the estimates of the parameters from Eqs. (4.2) and (4.4) after $k = 208$ iterations of the EM algorithm. Owing to an identification problem, we set the parameter $\tau = 2$.

Figure 4.2 about here

Besides parameter estimation the particle-filtering approach allows us to estimate the bubble process from the data. Figure 4.2 displays the estimated bubble process

[4]We used 500 particles and considered $l = 15$ lags.

(solid line) as well as the true (simulated) bubble process (taken from Figure 4.1). Obviously, the estimated bubble process almost perfectly fits the true bubble values. These econometrically reliable estimates of the bubble process can now be included in the information set Ω. This procedure ultimately enables us to analyze the conditional stock-price variance according to Eq. (4.10).

Figure 4.3 about here

Figure 4.3 displays the conditional stock-price variance (solid line) along with the estimated bubble process (dashed line). Two features are worth mentioning. (1) In line with our theoretical results from above, stock-price volatility collapses one period after the bubble burst. (2) The increase in the stock-price variance process in response to increases in the bubble process occurs with a considerable time delay. In Figure 4.3, for example, the bubble process begins to take on substantially increasing values around the date $t = 50$. By contrast, stock-price volatility remains (roughly) constant and begins to increase steadily no earlier than around the date $t = 110$.

To explain this latter phenomenon we again refer to the derivative in Eq. (4.11). For $B_{t-1} > \tau$ the slope of the stock-price variance essentially consists of the two summands within the squared brackets. For small and moderate values of B_{t-1} the first term is negligible and the slope of the stock-price variance is primarily determined by the constant value $\frac{2\kappa}{\psi}\left(\exp\left\{\iota^2\right\} - 1\right)$. During this period the stock-price variance remains largely unaffected by the B_{t-1}-levels. It is not until the bubble values get sufficiently large that the first summand (containing B_{t-1}) begins to dominate the slope of the stock-price variance triggering strongly increasing stock-price volatility.

The stock-price volatility dynamics displayed in Figure 4.3 is strongly consistent with the observations by Brunnermeier and Oehmke (2013) on the distinct volatility phases in the run-up to a financial crisis. If we interpret the start of the crisis as the first period after the burst of the stock-market bubble, their observation can be stated as follows. In an early stage when the bubble begins to emerge, stock-market volatility is comparably low. Then, volatility increases due to trading frenzy in a phase of euphoria which finally ends in the burst of the bubble. At the beginning of the financial crisis, that is at the moment of the crash, stock-market volatility is maximal.

4.4 Conclusion

This chapter analyzes (conditional) stock-price volatility dynamics in a present-value framework with periodically collapsing bubbles as proposed by Evans (1991). We derive closed-form expressions for the volatility paths and explore their properties theoretically. In an empirical part we describe a sequential Monte Carlo approach for extracting the unobservable bubble process from the data.

Our major finding is that the present-value framework produces stock-price volatility paths that are broadly consistent with empirically observed volatility structures in the run-up to financial crises and/or the burst of a stock-market bubble. Evans bubbles contribute to excessive stock-price volatility and volatility reaches its maximum when the bubble bursts. Our volatility results should be of interest to traders in international stock and derivative markets, for example for valuing stock-price sensitive claims.

Figures

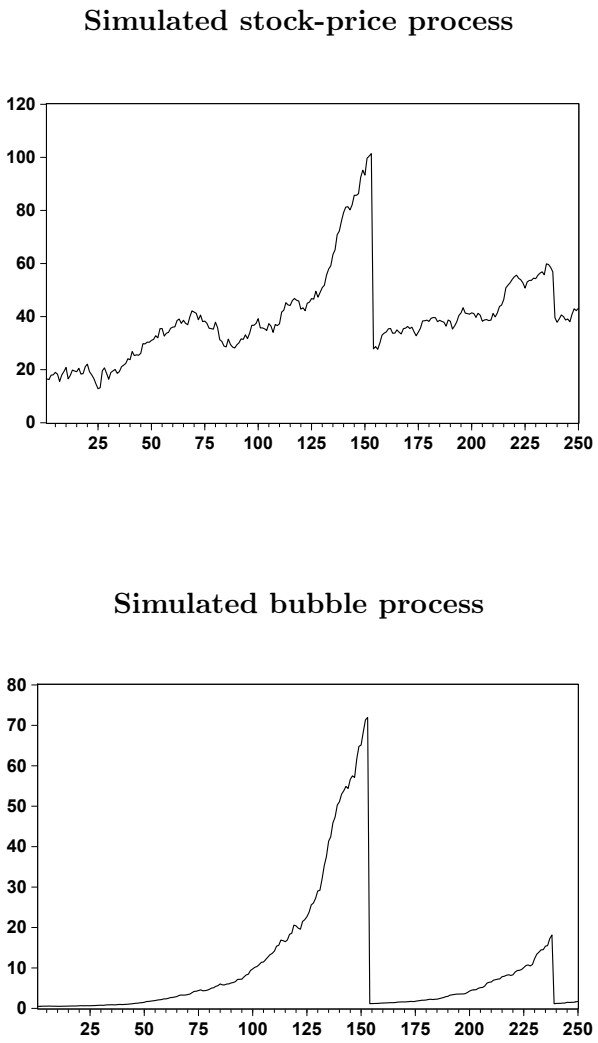

Figure 4.1: Stock-price process and included Evans bubble.

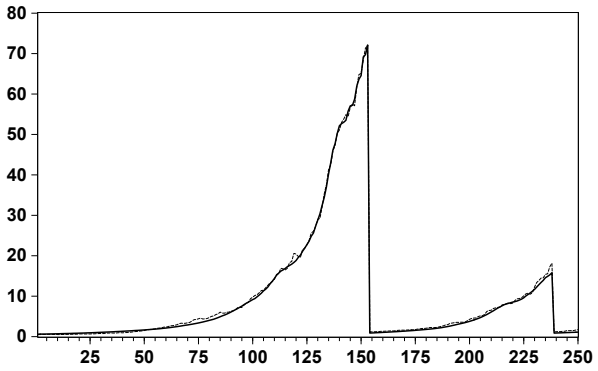

Figure 4.2: Estimated Evans-bubble process (solid line) versus true Evans-bubble process (dashed line).

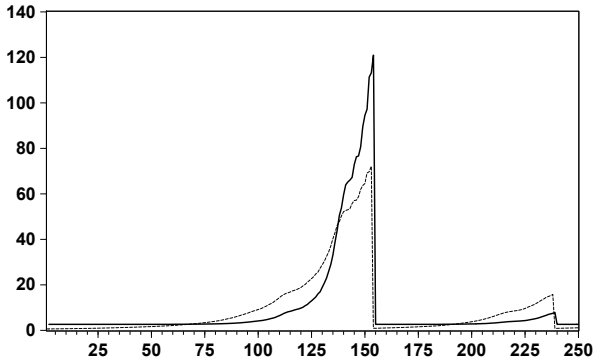

Figure 4.3: Conditional stock-price variance (solid line) and estimated Evans-bubble process (dashed line).

Tables

Parameter	True value	Estimate	Standard error
r	0.0200	0.0198	1.5783×10^{-8}
σ^2	0.0009	0.0010	9.3802×10^{-9}
ι^2	0.0010	0.0012	1.3541×10^{-6}
κ	1.1000	0.8776	0.0226
π	0.9800	0.9793	2.3847×10^{-5}

Table 4.1: Parameter estimates using the EM algorithm.

Chapter 5

Periodic and stochastically deflating rational bubbles

5.1 Introduction

The economic literature on stock-price bubbles provides a variety of econometric bubble specifications which more or less incorporate the theoretical characteristics of rational bubbles. As previously mentioned, the most influential model for rational bubbles is the periodically collapsing bubble proposed by Evans (1991). However, Evans' specification has two shortcomings. First, the Evans bubble always collapses entirely within one period. Second, the bubble always crashes to the same non-zero mean value. In view of the empirical results from Chapter 3, these theoretical features do not appear realistic when confronted with real-world stock-market data. Even though events of utmost importance to market participants might trigger substantial crashes, it is common sense that after a crash the adjustment of the stock price to its fundamental level follows a longer lasting process (see Allen and Gale, 2000). Furthermore, it is plausible that in the case of reoccurring bubbles the respective adjustment processes do not take the same amount of time. This appears obvious due to the differing characteristics of the associated phenomena that caused

the bubbles (see, Kindleberger and Aliber, 2005). Overall, these empirical and theoretical arguments suggest modeling the burst of a bubble with its subsequent adjustment process as a stochastically deflating process.

The purpose of this chapter is threefold. First, we present a new bubble model that is closely related to theory and to financial data. We establish a nonlinear model that is a combination of the periodically collapsing Evans bubble and the incompletely bursting bubble model proposed by Fukuta (1998). We show that our resulting bubble specification (a) is consistent with rational behavior, and (b) is periodically reoccurring with stochastically deflating adjustment processes. Second, we estimate our new (parametric) bubble specification using artificial and real-world stock-price data. As in Chapter 3, we perform estimation via a nonlinear state-space model using sequential Bayesian Monte Carlo methods. Third, we use the estimated latent bubble process to theoretically analyze conditional stock-price volatility. To this end, we establish a closed-form volatility formula that enables us to analyze the associated volatility dynamics analytically.

The structure of this chapter is as follows. Section 5.2 presents an overview of previous rational bubble specifications and introduces our new periodic, stochastically deflating bubble model. Section 5.3 contains the empirical application using artificial and real-world data. In Section 5.4 we establish the analytic stock-price volatility formula and analyze its dynamics. Section 5.5 concludes.

5.2 Alternative specifications of rational bubbles

Alternative specifications of rational bubbles within the present-value framework are scattered in the literature. These models are either deterministic or stochastic, but

have in common that they all satisfy the (discounted) martingale condition. Before introducing our new bubble model, we review some existing bubble specifications from the literature.

5.2.1 Previous rational bubble models

Invoking the martingale property Eq. (2.6), we may specify the simplest rational bubble as

$$B_t = \frac{1}{\psi} B_{t-1} + \omega_t, \tag{5.1}$$

where $0 < \psi < 1$ and ω_t is an i.i.d. error term with zero mean. This specification implies a continuous (stochastic) growth of the bubble. However, in view of Eq. (2.4) this process leads to infinitely large stock prices at some future time, a pattern inconsistent with real-world stock prices.

A more realistic bubble model is proposed by Blanchard (1979) and Blanchard and Watson (1982). The authors describe a bubble with two different states, namely

$$B_t = \begin{cases} \frac{1}{\pi\psi} B_{t-1} + \varpi_t & , \text{ with probability } \pi \\ \varpi_t & , \text{ with probability } 1 - \pi \end{cases}, \tag{5.2}$$

where ϖ_t is an i.i.d. error term with zero mean ensuring rationality. With probability π, the bubble grows at a higher than the required rate of return r, whereas the bubble can collapse with probability $1 - \pi$. In case of a collapse the bubble equals any realization of the stochastic error term ϖ_t. Although the bubble is rational and exhibits a periodically collapsing behavior, there are two drawbacks to mention. (1) According to the (discounted) martingale property from Eq. (2.6), Diba and Grossman (1988b) argued that, in general, a bubble cannot start from zero and negative bubbles are not possible if t tends to infinity (see Section 2.2). However,

owing to the error term ϖ_t, these two features are not excluded from the Blanchard bubble (5.2). (2) If the bubble collapses, it collapses more or less completely within one period and a slower or moderate deflation cannot occur.[1]

Figure 5.1 about here

The Evans (1991) bubble overcomes the fundamental critique by Diba and Grossman (1988b). Its econometric structure is described in Section 3.2. Up to date, the Evans bubble is considered to be the most realistic model and is therefore extensively referred to the literature. However, the Evans bubble has two shortcomings. Like the Blanchard bubble (5.2), the Evans bubble collapses completely within one period and furthermore always drops back to the same non-zero mean constant (κ). This behavior is far from reality, since the crises of the last century showed that adjustments typically take more time than one time unit. To state an example, the Dot-com bubble deflated in the period between March 2000 and October 2001 during which the NASDAQ lost over 70% of its peak value (see Figure 5.1). In similar vein, caused by the subprime mortgage crisis in 2007 the Dow Jones Industrial Average lost more than 50% of its value between October 2007 and March 2009. These bubble deflations always started with bad news to investors under market uncertainty and panic lead to an acceleration of the adjustment process. However, such downturns may sometimes be interrupted, for example, by the intervention of central banks providing room for a short-term recovery of prices. Furthermore, it is obvious that the distinct adjustment processes are likely to differ from each other. For example, Figure 5.1 reveals that the adjustment of the NASDAQ between March 2000 and October 2001 differs from the deflating process between October 2007 and March 2009.

[1]The exact value of the bubble reduction is driven by the error term ϖ_t and the probability π.

An alternative model that is able to account for a more flexible deflating behavior is proposed by Fukuta (1998). The author establishes a so-called incompletely bursting bubble that generalizes the Blanchard and the Evans bubble. The formal specification consists of three possible states:

$$B_t = \begin{cases} \frac{1}{\psi}\frac{\alpha_1}{\pi_1}B_{t-1} & \text{, with probability } \pi_1 \\ \frac{1}{\psi}\frac{\alpha_2}{\pi_2}B_{t-1} & \text{, with probability } \pi_2 \\ \frac{1}{\psi}\frac{1-\alpha_1-\alpha_2}{1-\pi_1-\pi_2}B_{t-1} & \text{, with probability } 1-\pi_1-\pi_2 \end{cases}, \tag{5.3}$$

where $(0 < \alpha_1 < 1)$, $(0 < \alpha_2 < 1)$ and $(0 < 1 - \alpha_1 - \alpha_2 < 1)$. The strictly positive probabilities of each state are given by π_1, π_2 and $1 - \pi_1 - \pi_2$. In addition, it is assumed that $(1-\alpha_1-\alpha_2)/(1-\pi_1-\pi_2) < \alpha_2/\pi_2 < \alpha_1/\pi_1$. It can be shown that in state 1 and 2, we have $B_t > B_{t-1}$, while in state 3 we have $B_t < B_{t-1}$. As a result, this specification consists of a large bubble state (state 1), a small bubble state (state 2) and an incompletely bursting state (state 3). However, an important drawback of this model is that within each state the bubble is subject to deterministic growth.

5.2.2 A new model for rational bubbles

In this section we introduce a new rational bubble specification which is strictly positive, periodic and stochastically deflating. Technically, we achieve this by combining features of the Evans bubble with those of the Fukuta model. Our specification consists of two different states, namely

$$B_t = \begin{cases} \frac{\alpha}{\psi\pi}B_{t-1}u_t & \text{, with probability } \pi \\ \frac{1-\alpha}{\psi(1-\pi)}B_{t-1}u_t & \text{, with probability } 1-\pi \end{cases}, \tag{5.4}$$

which can more compactly be written in one equation as

$$B_t = \Big[\Big(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)\nu_t + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}\Big]u_t. \tag{5.5}$$

As in Fukuta (1998), the parameter α is arbitrary with $0 < \alpha < 1$. This constraint ensures that the bubble never collapses to zero and thus can rebuild. As in Evans (1991) ν_t is an i.i.d. Bernoulli distributed process with $\Pr(\nu_t = 1) = \pi$ and $\Pr(\nu_t = 0) = 1 - \pi$ for $0 < \pi \leq 1$. The error term u_t is assumed to be an i.i.d. lognormally distributed random process, satisfying $u_t > 0$ and $E_{t-1}(u_t) = 1$ for all t, that is $u_t = \exp(y_t - \iota^2/2)$ with $y_t \sim N(0, \iota^2)$. In addition, the covariance between ν_t and u_t is assumed to be zero. The lognormal random variable u_t ensures a stochastic bubble growth and a strictly positive bubble process. Moreover, this random variable also lead to huge upward and downward movements of the bubble process depending on the variance parameter ι^2.

Two further assumptions of the model are $\frac{\alpha}{\pi} > 1$ and $\frac{1-\alpha}{1-\pi} < \psi$ which guarantee two certain states. In state 1 the bubble grows at a mean rate $\frac{\alpha}{\psi\pi}$, that is at a higher rate than the required rate of return, but the bubble is subject to a possible burst. State 2 models the burst of the bubble. In this state the bubble deflates because it evolves at a mean rate $\frac{1-\alpha}{\psi(1-\pi)} < 1$. The probability of this deflating burst is $1 - \pi$. Depending on the explicit parameter constellation, these bursts can be more or less extensive ranging from *small corrections* to *big crashes* within one or several periods. Our specification bears some similarity to the Fukuta model with only two states, but the additional error term u_t in Eq. (5.5) allows for a stochastic and more realistic growth/decay within each state. The formal structure of our model is more parsimonious than the Fukuta and Evans specifications, but more flexible in describing real-world bubble processes.

It remains to prove that our bubble specification (5.5) is consistent with rational behavior. To verify the martingale property, we need the conditional expectation

$$E_{t-1}\left(B_t|B_{t-1}\right) = E_{t-1}\left(\Big[\Big(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)\nu_t + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}\Big]u_t|B_{t-1}\right). \quad (5.6)$$

Owing to the independence of ν_t and u_t and the definition of u_t, we have

$$\begin{aligned} E_{t-1}\left(B_t|B_{t-1}\right) =& E_{t-1}\left(\Big(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)\nu_t + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}|B_{t-1}\right) E_{t-1}\left(u_t|B_{t-1}\right) \\ =& E_{t-1}\left(\Big(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)\nu_t + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}|B_{t-1}\right). \end{aligned}$$

The Bernoulli distributed random variable ν_t equals 1 with probability π and 0 with probability $1-\pi$, so that

$$\begin{aligned} E_{t-1}\left(B_t|B_{t-1}\right) =& E_{t-1}\left(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)} + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}|B_{t-1}, \nu_t = 1\right) \\ &+ E_{t-1}\left(\Big(\frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}|B_{t-1}, \nu_t = 0\right), \end{aligned}$$

implying

$$\begin{aligned} E_{t-1}\left(B_t|B_{t-1}\right) =& \pi\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)} + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1} + (1-\pi)\Big(\frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1} \\ =& \frac{\alpha}{\psi}B_{t-1} + \frac{1-\alpha}{\psi}B_{t-1} \\ =& \frac{1}{\psi}B_{t-1}, \end{aligned} \quad (5.7)$$

which verifies the martingale property (2.6).

Figure 5.2 about here

Figure 5.2 displays four trajectories of our stochastically deflating bubble process (5.5). For all simulations we set $\psi = 0.9804$, but choose different values for ι^2, α and π. We initiate the bubble processes with $B_0 = 0.5$. The trajectories consist of 250

observations representing a timespan of approximately 21 years based on monthly data.

All four trajectories in Figure 5.2 exhibit two or three major bubbles. All bubbles differ in their deflating structures with respect to the quantitative degree and the speed of deflating. The emergent phases of these bubbles are generally accompanied by explosive growing behavior. After reaching the peak value, the bubble processes may deflate either quickly or moderately/slowly over longer lasting periods. An apparent feature of the trajectories are the stochastic fluctuations both during the emergent phase (stochastic bubble growth) as well as during the adjustment process (stochastic deflating).

5.3 Estimating periodic, stochastically deflating bubbles via particle-filter methods

In this section we use the particle-filter methods, presented in Sections 3.3 and 3.4 to estimate our new bubble specification from the data. In a first step, we use an artificial data set to check for the reliability of the estimation methodology, while in a second step, we apply the technique to real-world stock-price and dividend data.

5.3.1 Nonlinear state-space representation

As in Section 3.2 we invoke the linear present-value model according to which the real stock price at time t, P_t, is given by

$$P_t = \phi \cdot D_t + B_t + \varepsilon_t. \tag{5.8}$$

As before, D_t denotes the real dividend payment and ε_t is a Gaussian white-noise error term with variance σ_ε^2. Thus, the fundamental value of the stock price is given by $\phi \cdot D_t + \varepsilon_t$ where ε_t reflects all other fundamentals not captured by the dividends. B_t is the latent bubble component and now specified according to our new bubble model as

$$B_t = \Big[\Big(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)\nu_t + \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}\Big]u_t. \tag{5.9}$$

Again, Eq. (5.8) is the observation equation and Eq. (5.9) the state equation of the nonlinear state-space model.

Next, we need the conditional distributions of the equations in order to apply the particle-filter methods. First, the distribution of the observation equation is Gaussian:

$$p(P_t \mid B_t, D_t; \boldsymbol{\theta}) \sim N\Big((\phi D_t + B_t), \sigma_\epsilon^2\Big). \tag{5.10}$$

Second, the conditional distribution of the state equation is given by the distribution of our new bubble model. Owing to the Bernoulli process, the distribution is a mixture of two lognormal distributions and given by

$$\begin{aligned} p(B_t \mid B_{t-1}; \boldsymbol{\theta}) \quad \sim \quad & \pi \cdot LN\Big(\frac{-\iota^2}{2} + \log\Big(\frac{\alpha}{\pi\psi}B_{t-1}\Big), \iota^2\Big) \\ & +(1-\pi) \cdot LN\Big(\frac{-\iota^2}{2} + \log\Big(\frac{1-\alpha}{(1-\pi)\psi}B_{t-1}\Big), \iota^2\Big). \end{aligned} \tag{5.11}$$

We collect all model parameter in the vector $\boldsymbol{\theta} = (\phi, \sigma_\varepsilon^2, \psi, \iota^2, \pi, \alpha)$.

Figure 5.3 about here

5.3.2 Artificial data

Figure 5.3 displays the trajectories of the economic framework. The stock price is calculated by using the price equation (5.8) whereas the dividends are generated by a simple random walk. The bubble is generated by Eq. (5.9) and we use the realization of the deflating bubble process given in the upper left graphic of Figure 5.2. The parameter vector of the entire nonlinear state-space model is set to be $\boldsymbol{\theta} = (\phi = 50, \sigma_\varepsilon^2 = 1.5, \psi = 0.9804, \iota^2 = 0.02, \pi = 0.87, \alpha = 0.91)$. As before, the artificial dataset consists of 250 observations.

Table 5.1 about here

5.3.3 Estimation results

Prior to estimating the latent bubble process, we have to identify our nonlinear state-space model using the particle based EM algorithm. Table 5.1 displays the estimates of the parameter vector $\boldsymbol{\theta}$ from the Eqs. (5.8) and (5.9). We used a similar estimation procedure as in Section 3.5.3 with $N = 300$ particles and for the convergence criterion we used $c = 1/300$. The EM algorithm was initiated by the vector $\boldsymbol{\theta_0} = (\phi = 30, \sigma_\varepsilon^2 = 0.5, \psi = 0.85, \iota^2 = 0.01, \pi = 0.7, \alpha = 0.75)$. We numerically maximized the Q-function from Eqs. (3.30) and (3.31) by the use of the FMINCON module of the software package MATLAB. Due to the convergence criterion the algorithm stopped after k = 285 iterations. The standard errors were computed from the Duan and Fulop (2011) stable estimator of the information matrix.[2]

[2]We used 500 particles and we considered $l = 15$ lags.

All parameters are identified and significant at all conventional levels.[3] Except for the variance parameters σ_ε^2 and ι^2, all other parameters are accurately estimated. Moreover, the unrestricted estimates $\hat{\alpha}$, $\hat{\psi}$ and $\hat{\pi}$ satisfy the model constraints $\frac{\alpha}{\pi} > 1$ and $\frac{1-\alpha}{1-\pi} < \psi$.

Figure 5.4 about here

Figure 5.4 displays the true bubble process (dashed lines) and the filtered bubble process obtained on the basis of parameter estimates from Table 5.1 and by using the particle filter and the particle smoother (solid lines). In both cases we used $N = 500$ particles. Obviously, the true bubble process is almost perfectly estimated enabling us to perfectly identify all highly speculative periods, bursts and deflating periods. It is remarkable that we are able to detect almost all movements of the bubble component, although our new bubble specification is much more volatile than the Evans bubble.

5.3.4 Real-world data

We use the same data set as described in Section 3.5.5. To recall, we consider real prices and real dividends of four stock-price indices, namely the DAX, the S&P 500, the NASDAQ and the HSI for the period between January 1981 and February 2014. Each time series consists of 398 observations. The trajectories of the data are displayed in Figure 3.6.

Table 5.2 about here

[3] In contrast to the estimation of the Evans model (cf. Section 3.5.3), all parameters are identified. This econometric improvement is due to our more parsimonious bubble specification.

5.3.5 Estimation results

Table 5.2 contains the parameter estimates for the four stock-price indices. The estimations were run with $N = 300$ particles and we used $c = 1/300$ for our convergence criterion. In case of no convergence, the estimation was terminated after 500 iterations. An appropriate initial parameter vector $\boldsymbol{\theta_0}$ for the EM algorithm was chosen using the results of Chapter 3. In particular, for each stock index the initialized value of the parameters ϕ, σ_ε^2, ψ and ι^2 were chosen close to the corresponding estimates from Section 3.5.6. The starting values for π and α were chosen as 0.7 and 0.75. The EM algorithm stopped after $k = 256$ iterations for the DAX and after $k = 219$ iterations for the HSI. For the NASDAQ and S&P 500 we terminated the algorithm after $k = 500$ iterations.[4]

All parameters are identifiable and significant. The estimates satisfy the theoretical bubble constraints $\frac{\alpha}{\pi} > 1$ and $\frac{1-\alpha}{1-\pi} < \psi$ implying two certain states. In state 1, the bubble grows at a higher mean rate than the required rate of return, while the bubble deflates in state 2. The estimated discount factors are smaller than one and imply a required rate of return of 0.89% on the DAX, 1.63% on the NASDAQ, 1.54% on the S&P 500 and 4.29% on the HSI. It follows from Eq. (2.6) and Eq. (5.7) that all estimated bubbles are rational.

The bursting (deflating) probabilities (given by $\Pr(\nu_t = 0) = 1 - \pi$) are higher for the DAX and the NASDAQ than for the S&P 500 and HSI implying that the deflating state is more likely to realize for the first two indices than for the latter two. The factor $\frac{1-\alpha}{(1-\pi)\psi}$, representing the mean rate of adjustment in case of a bubble burst, is 0.1159 for the HSI indicating much more pronounced bursts for the HSI

[4] Although the parameters stopped changing substantially after a few hundred iterations, the EM algorithm did not converge.

than for the other indices. This result is further supported by the relatively high estimated variance parameter $\hat{\iota^2}$ of the lognormal error term. For the DAX and NASDAQ the mean rates are equal to 0.7782 and 0.8155, respectively, indicating more moderate adjustments. The mean rate of 0.5626 for the S&P 500 points to an intermediate bubble adjustment. We conclude that a bubble burst or deflating is likely to occur more frequently for the DAX and NASDAQ. However, if the bubble bursts, the adjustments tend to be extreme for the HSI, whereas for the DAX, NASDAQ and S&P 500 the bubbles deflate with a moderate mean rate.

Figure 5.5 about here

Figure 5.5 displays the estimated bubble components of the four price indices extracted by the particle smoother (with $N = 500$ particles) and the corresponding fundamental processes of the four indices. As in Chapter 3, our bubble paths still evolve in accordance with the corresponding stock-price paths. However, the dynamic relationship between bubble and stock-price values appears to have decreased considerably since the fundamental processes have become more important than under the Evans bubble. This increased importance of the fundamental process is reflected by the estimates $\hat{\phi}$ which are larger now than under the Evans bubble.

The bubble processes for the DAX, NASDAQ and S&P 500 in Figure 5.5 all exhibit two extensively deflating periods, namely (a) during the new-economy crisis (2000-2003), and (b) during the subprime mortgage and credit crisis (2007-2008). We further find moderate adjustment phases (lasting one or two months) for the DAX in the aftermath of the Black Monday (October 1987). Although our bubble trajectories exhibit several bursts, they never deflate completely and restart from different values. The HSI bubble process exhibits several pronounced peaks. How-

ever, the adjustment phases of the HSI bubble are typically more pronounced and shortly lived. Apart from the deflating period between the end of 2000 and mid-2003, the bubble always deflate within a few months. Moreover, the HSI bubble peaks often deflate almost completely and the bubble restart to emerge from values near zero.

Figure 5.6 about here

Figure 5.6 displays the ratios of the filtered bubble components and corresponding the stock-price series. In January 1981 40% of the DAX value could be attributed to a bubble component, whereas for the NASDAQ and S&P 500 the ratio was near zero but growing to approximately 70% for the NASDAQ and 44% for the S&P 500 in the run-up to the Black Monday (October 1987). For these three indices, the ratios peaked at the beginning of the new-economy crisis. In particular the DAX ratio peaked in February 2000 (84%), the NASDAQ ratio in September 2000 (94%) and the S&P 500 ratio in August 2000 (74%). Compared to these three indices, the HSI ratio remains relatively small and ranges between 0% and 60%, but shows several pronounced adjustments during the observation period.

Figure 5.7 about here

Table 5.3 about here

5.3.6 Model diagnostics

We now use the goodness-of-fit tests from Section 3.5.4 to check how well our new bubble specification fits the real-world data. Figure 5.7 displays the respective cdfs

of the PITs for the four stock indices considered. The corresponding confidence intervals were computed using the critical value at the 5% level of the KS test (0.6808). Obviously, the cdfs of the PITs are all close to the 45-degree line. For the DAX and the NASDAQ the KS test indicates no significant differences of the PITs from the U(0,1) distribution at the 10% level (see Table 5.3). However, for the S&P 500 and HSI indices there are significant deviations from the U(0,1) distribution at the 5% and 1% levels, respectively. Evidently, the i.i.d. assumption is violated for all four indices as indicated by the LB tests displayed in Table 5.3. The diagnostic tests seem to indicate some kind of misspecification of our new bubble process when fitted to real-world data.

5.4 Volatility analysis

Finally, we analyze the theoretical (conditional) stock-price volatility within the present-value framework when including our new periodic, stochastically deflating bubble. The aim of this section is to compare the theoretical stock-price volatility induced by our new bubble specification with the volatility structure induced by the Evans bubble as described in Chapter 4.

5.4.1 Conditional stock-price volatility

Following the lines of Chapter 4, it is straightforward to find the variance of the stock price P_t conditional on all information available to market participants as of date $t-1$. In the present-value model with rational expectations, this conditional stock-price variance is generally given by

$$\mathrm{Var}_{t-1}(P_t) = \mathrm{Var}_{t-1}(B_t) + \beta^2\sigma^2, \tag{5.12}$$

where ϵ_t is an i.i.d. Gaussian white-noise process with mean zero and variance σ^2 and $\beta = 1/r$.[5] In contrast to Chapter 4, we now specify the bubble term B_t in Eq. (5.12) as our new bubble model given in Eq. (5.5).

To obtain a closed-form volatility formula of the stock price, we have to find an analytic expression for $\mathrm{Var}_{t-1}(B_t)$ in Eq. (5.12). Using the distribution result from Eq. (5.11), we have

$$\begin{aligned}
\mathrm{Var}_{t-1}(B_t) &= \mathrm{Var}_{t-1}\left(\Big(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}\nu_t u_t + \Big(\frac{1-\alpha}{\psi(1-\pi)}\Big)B_{t-1}u_t\right) \\
&= \left(\frac{\alpha B_{t-1}}{\psi\pi} - \frac{(1-\alpha)B_{t-1}}{\psi(1-\pi)}\right)^2 \cdot \mathrm{Var}_{t-1}(\nu_t u_t) \qquad (5.13) \\
&\quad + \left(\frac{(1-\alpha)B_{t-1}}{\psi(1-\pi)}\right)^2 \cdot \mathrm{Var}_{t-1}(u_t) \\
&\quad + 2\left(\frac{\alpha}{\psi\pi} - \frac{1-\alpha}{\psi(1-\pi)}\right) \cdot \left(\frac{1-\alpha}{\psi(1-\pi)}\right) \cdot B_{t-1}^2 \cdot \mathrm{Cov}_{t-1}(u_t, \nu_t u_t).
\end{aligned}$$

The conditional covariance $\mathrm{Cov}_{t-1}(u_t, \nu_t u_t)$ and the conditional variances $\mathrm{Var}_{t-1}(u_t)$ and $\mathrm{Var}_{t-1}(\nu_t u_t)$ can be computed from the distributional assumptions on the error terms u_t and ν_t. Overall, we obtain

$$\begin{aligned}
\mathrm{Var}_{t-1}(B_t) &= \left(\frac{\alpha B_{t-1}}{\psi\pi} - \frac{(1-\alpha)B_{t-1}}{\psi(1-\pi)}\right)^2 \cdot \left(\exp\{\iota^2\}\pi - \pi^2\right) \\
&\quad + \left(\frac{(1-\alpha)B_{t-1}}{\psi(1-\pi)}\right)^2 \cdot \left(\exp\{\iota^2\} - 1\right) \\
&\quad + 2\left(\frac{(\alpha-\pi)(1-a)}{\psi^2\pi(1-\pi)^2}\right) B_{t-1}^2 \cdot \left[\pi\left(\exp\left\{\iota^2\right\} - 1\right)\right]. \qquad (5.14)
\end{aligned}$$

Finally, inserting Eq. (5.14) into Eq. (5.12), the conditional stock-price variance is given by

$$\mathrm{Var}_{t-1}(P_t) = \left(\frac{(\alpha-\pi)^2}{\psi\pi(1-\pi)^2}\right) \cdot \left(\exp\{\iota^2\} - \pi\right) B_{t-1}^2$$

[5] For a detailed derivation see Section 4.2.

$$+\left(\frac{(1-\alpha)^2+2(\alpha-\pi)(1-\alpha)}{\psi^2(1-\pi)^2}\right)\cdot\left(\exp\{\iota^2\}-1\right)B_{t-1}^2$$
$$+\beta^2\sigma^2. \tag{5.15}$$

5.4.2 Volatility dynamics

Similar to the results from Chapter 4, we conclude from Eq. (5.15) that the presence of the new bubble type necessarily increases stock-price volatility as long as the parameter constraints $0<\alpha,\pi<1$ and $\frac{\alpha}{\pi}>1$ are satisfied. This may be confirmed analytically by considering the derivative

$$\begin{aligned}\frac{\partial \mathrm{Var}_{t-1}(P_t)}{\partial B_{t-1}} &= B_{t-1}\cdot\left[\left(\frac{2(\alpha-\pi)^2}{\psi\pi(1-\pi)^2}\right)\cdot\left(\exp\{\iota^2\}-\pi\right)\right.\\ &\quad\left.+\left(\frac{2(1-\alpha)^2+4(\alpha-\pi)(1-\alpha)}{\psi^2(1-\pi)^2}\right)\cdot\left(\exp\{\iota^2\}-1\right)\right]. \end{aligned}\tag{5.16}$$

Since $\alpha,\pi<1$ and $\alpha>\pi$, this derivative indicates a strictly positive relationship between the infinitesimal change in stock-price volatility and B_{t-1}. Furthermore, the conditional stock-price variance given in Eq. (5.15) is a function of B_{t-1} so that stock-price volatility collapses one period after the burst of the bubble. As a result, we find that our new periodic, stochastically deflating bubble entails the same relationship between the bubble component and stock-price volatility as the Evans bubble.

To get further insights into the volatility dynamics, we compute a concrete volatility path generated by Eq. (5.15). As in Chapter 4, we consider the stock-price equation (4.4) with $\beta=1/r=50$, and assume that dividends follow a driftless random walk with variance $\sigma^2=0.0009$. As the bubble component we use the stochastically deflating bubble displayed in Figure 5.3 with parameters $\psi=0.9804$, $\iota^2=0.02$, $\pi=0.87$ and $\alpha=0.91$. Furthermore, we assume this latent bubble

process to be known, so that it is directly included in the associated information set Ω of $\text{Var}_{t-1}(P_t)$.[6]

Figure 5.8 about here

Figure 5.8 displays the conditional stock-price variance (solid line) along with the new bubble process (dashed line). First, we note that the more realistic deflating behavior directly transfers to the volatility process. Second, in line with the results from above, the stock-price volatility process attains its maximal values when B_{t-1} takes on its largest values which typically occurs on the eve of the bubble crash. Third, stock-price volatility starts to deflate one period after the bubble, which is visible for the three bubble peaks (thin lines). Finally, there is a time delay between an increase in the variance process and an increase in the bubble process. However, this delay is not as substantial as in the Evans-bubble process.

Overall, stock-price volatility dynamics under a periodic, stochastically deflating bubble is in line with the volatility results established in Chapter 4 under the Evans bubble. We find the same theoretical characteristics of the conditional stock-price variance implying that the volatility dynamics is still consistent with that described by Brunnermeier and Oehmke (2013) and the Minsky model.

5.5 Conclusion

In this chapter we establish a new parametric specification for rational bubbles, the so-called periodic, stochastically deflating bubble. This model is a combination of

[6] We show in Sections 4.3.2 and 5.3.3 that the particle-filter methods enable us to estimate the latent bubble process from the data. However, in this section we use the simulated bubble process directly and refrain from any further estimation.

the well-known Evans (1991) bubble and the incompletely bursting bubble model proposed by Fukuta (1998). Our new model incorporates all properties of the famous Evans bubble, but is empirically more plausible. In contrast to the Evans bubble, our specification allows the bubble to deflate over more than one period and to restart from different stock-price values at the end of the adjustment process.

In the empirical part of this chapter we estimate our new bubble specification by particle-filter techniques within a state-space framework using artificial as well as real-word stock-price and dividend data. In contrast to Evans' model, all parameters of our bubble specification are empirically identifiable by our estimation methodology. The estimation results for the real-world stock-price indices explicitly indicate the presence of a periodic and stochastically deflating bubble in the data.

Diagnostic specifications based on the PIT approach proposed by Diebold et al. affirm a good distributional fit of the bubble specification to the data. However, there are still some specification shortcomings (like existing autocorrelation among the PITs) that have to be overcome in future research.

Figures

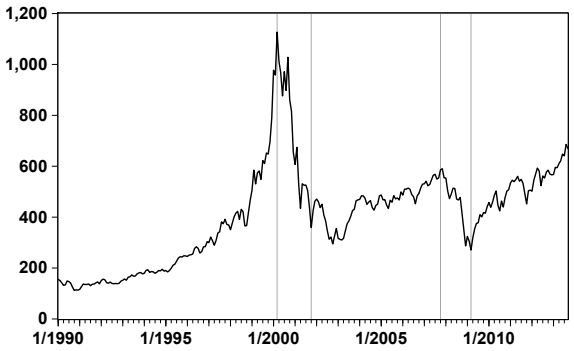

Figure 5.1: NASDAQ stock-market index, January 1990 - October 2013.

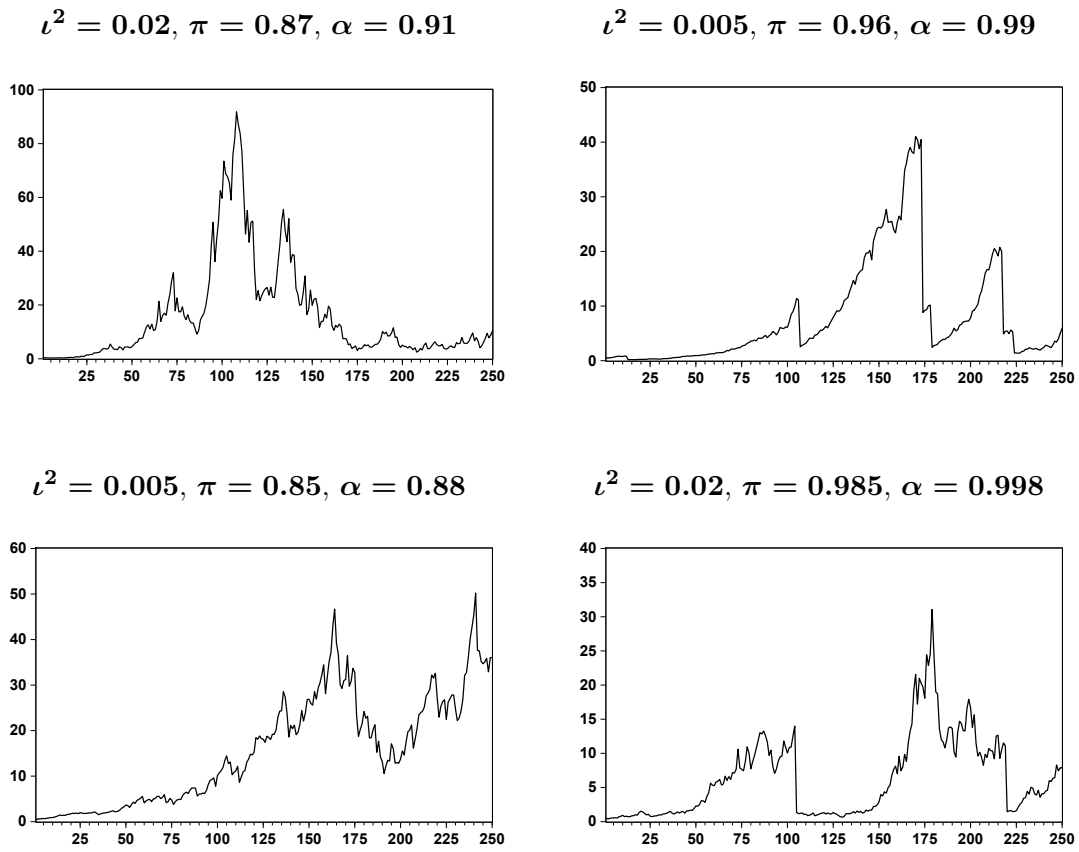

Figure 5.2: Simulated stochastically growing and deflating bubble trajectories according to Eq. (5.5).

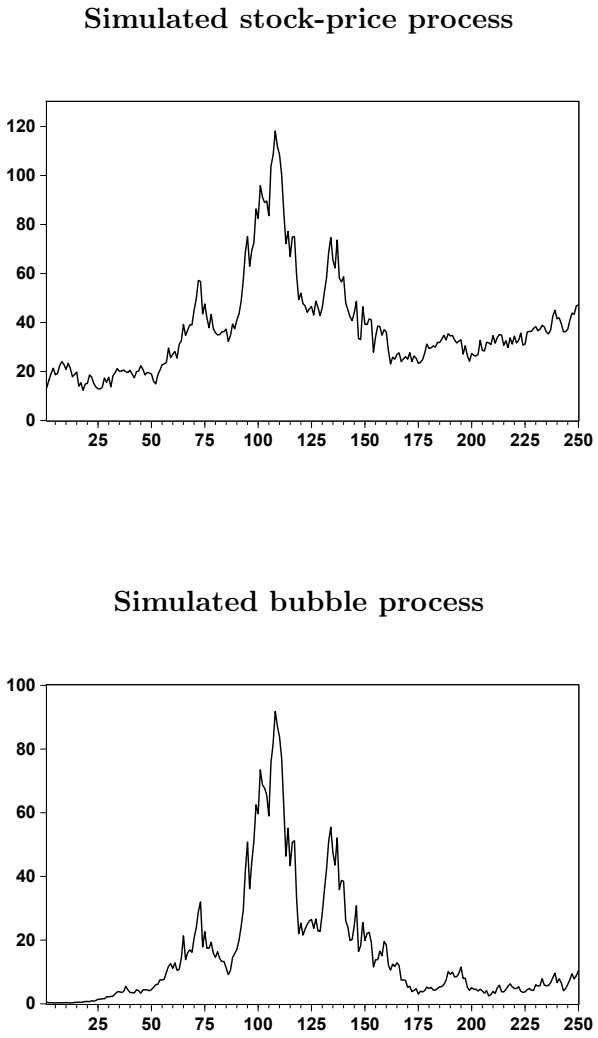

Figure 5.3: Stock-price process and included periodic, stochastically deflating bubble.

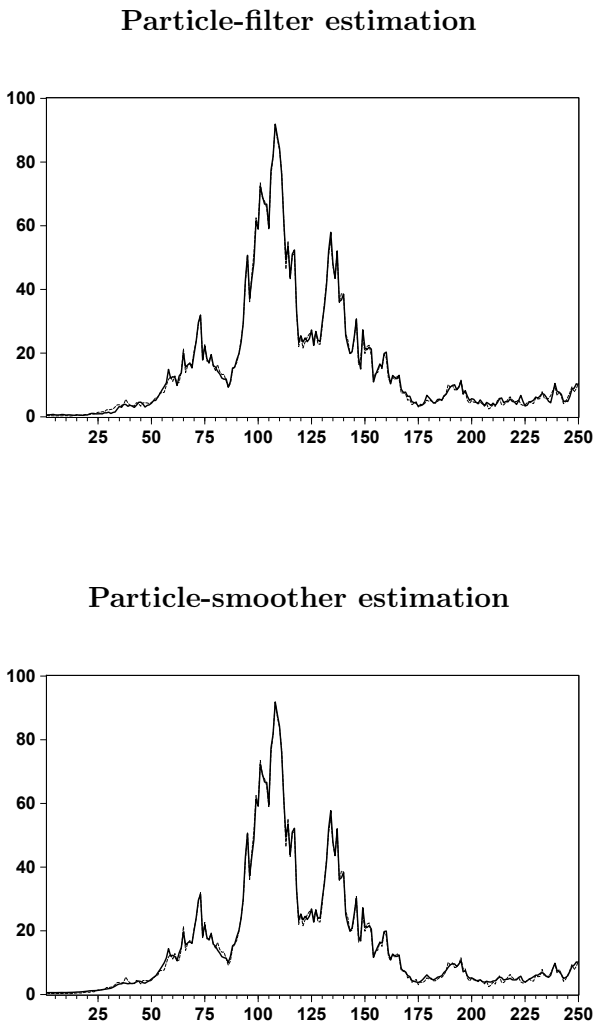

Figure 5.4: Estimated bubble process (solid lines) vs. true bubble process (dashed lines).

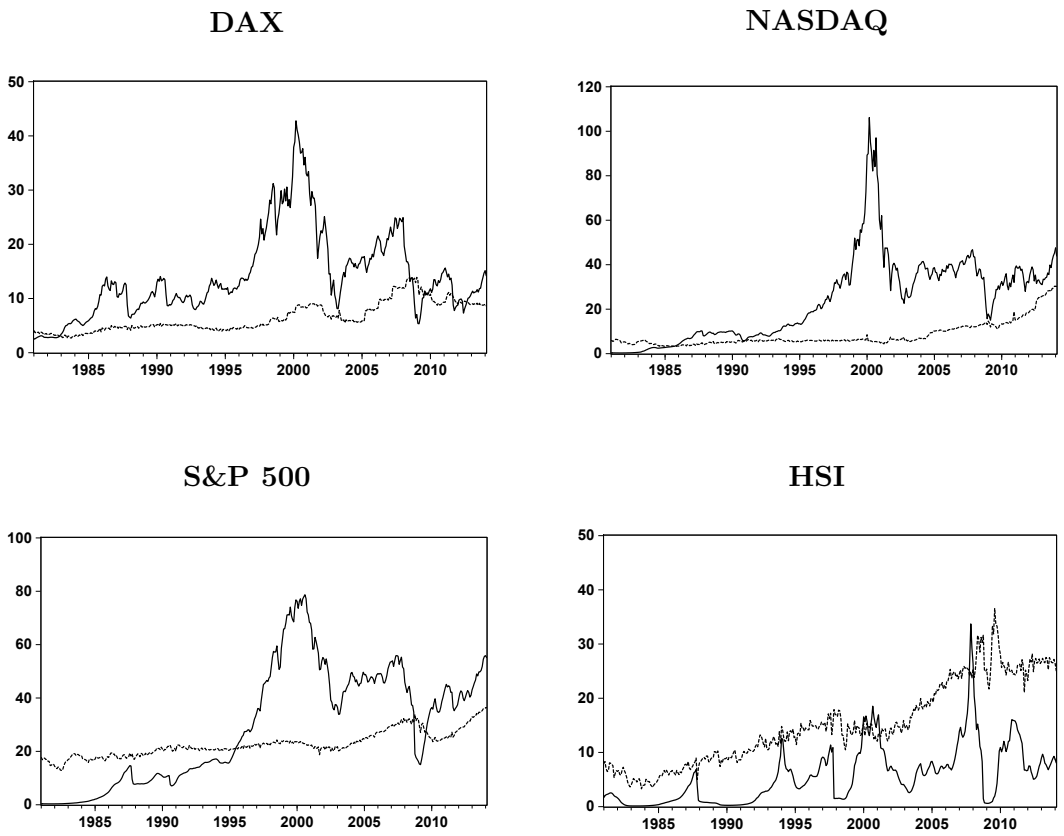

Figure 5.5: Estimated periodic, stochastically deflating bubble processes (solid lines) and the fundamental processes (dashed lines) for the DAX, NASDAQ, S&P 500 and HSI.

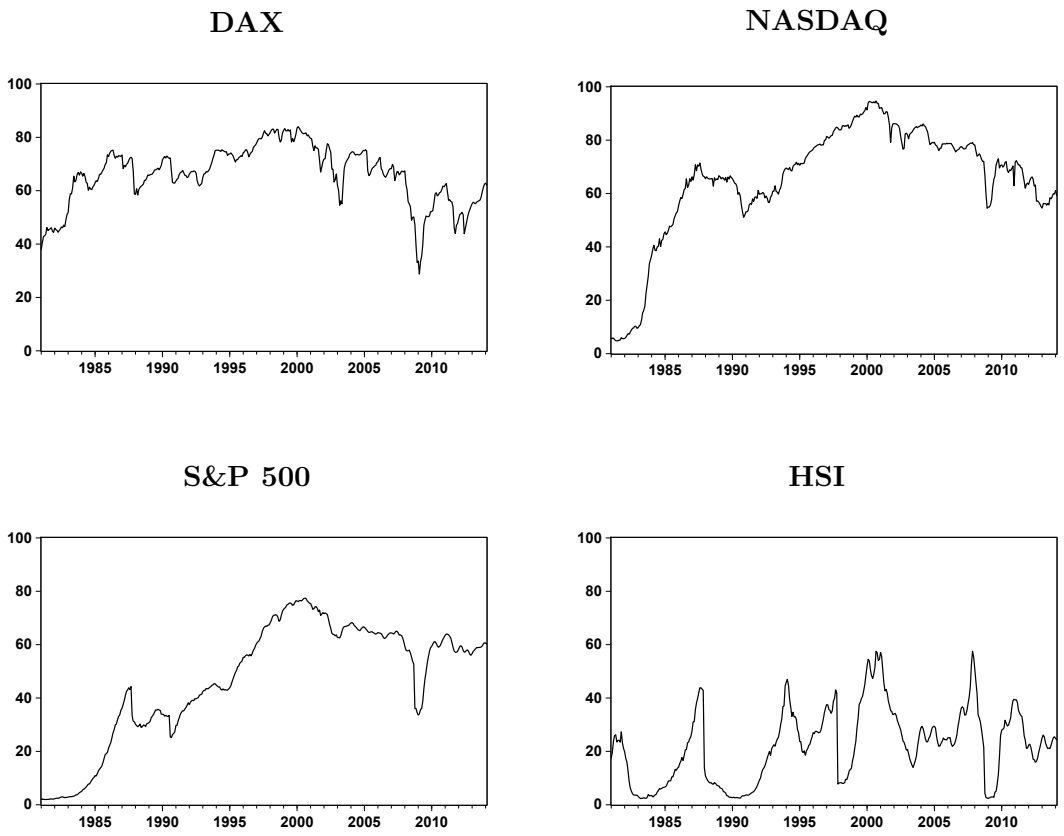

Figure 5.6: Ratios of the periodic, stochastically deflating bubble components and the stock-price series for the DAX, NASDAQ, S&P 500 and HSI.

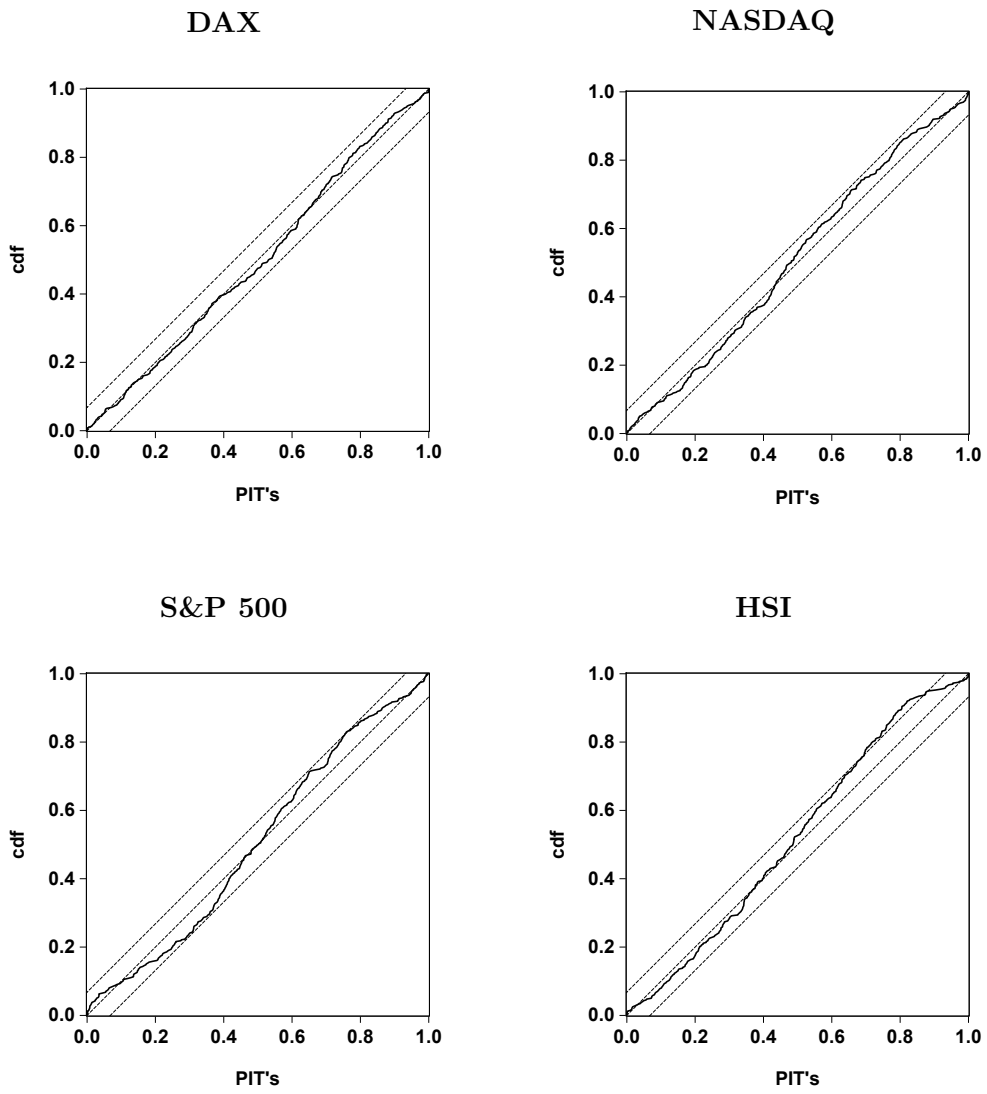

Figure 5.7: Cdfs of the PITs for the DAX, NASDAQ, S&P 500 and HSI. True cdfs and 5% confidence intervals are represented by dashed lines.

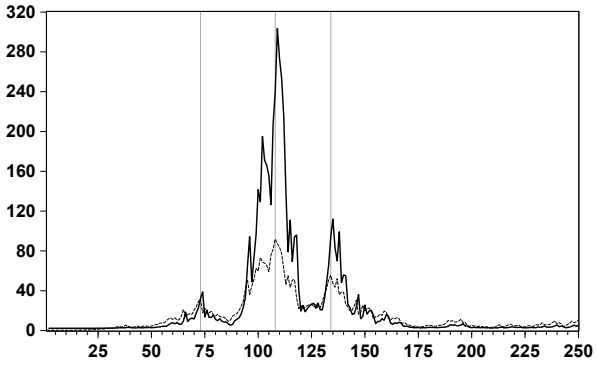

Figure 5.8: Conditional stock-price variance (solid line) and periodic, stochastically deflating bubble process (dashed line).

Tables

Parameter	True	Estimate	Standard error
ϕ	50.0000	49.8220	0.5112
σ_ε^2	1.5000	1.6839	0.2072
ψ	0.9804	0.9698	1.6239×10^{-4}
ι^2	0.0200	0.0231	6.2637×10^{-5}
π	0.8700	0.8917	0.0073
α	0.9100	0.9255	0.0044

Table 5.1: Parameter estimates using the EM algorithm.

Parameter	DAX	NASDAQ	S&P 500	HSI
ϕ	13.3782	29.9531	20.2057	23.0731
	(0.0308)	(0.0324)	(0.0037)	(0.0101)
σ_ε^2	0.1277	0.4476	1.3530	3.1976
	(3.3094×10^{-4})	(0.0014)	(0.0033)	(0.0109)
ψ	0.9912	0.9840	0.9848	0.9589
	(2.0210×10^{-5})	(1.7480×10^{-5})	(9.8682×10^{-6})	(1.8464×10^{-4})
ι^2	0.0047	0.0061	0.0036	0.0414
	(2.2588×10^{-7})	(7.6905×10^{-7})	(6.7574×10^{-8})	(5.5004×10^{-5})
π	0.9484	0.9595	0.9926	0.9865
	(2.2129×10^{-4})	(0.0011)	(3.7926×10^{-5})	(6.2247×10^{-5})
α	0.9602	0.9675	0.9959	0.9985
	(1.4712×10^{-4})	(8.6242×10^{-4})	(1.3766×10^{-5})	(9.0434×10^{-7})
$\frac{\alpha}{\pi}$	1.0124	1.0083	1.0033	1.0122
$\frac{1-\alpha}{1-\pi}$	0.7713	0.8024	0.5540	0.1111

Table 5.2: Results of the parameter estimation for the DAX, NASDAQ, S&P 500 and HSI. Standard errors are in parentheses.

Test statistic	DAX	NASDAQ	S&P 500	HSI
KS test	0.0394	0.0560	0.0708**	0.0998***
LB test	31.6201**	86.6006***	385.4446***	335.5379***

Table 5.3: KS tests and LB tests on the PITs of the DAX, NASDAQ, S&P 500 and HSI. ***, **, * denote statistical significance at 1%, 5% and 10% levels, respectively.

Chapter 6

Summary and outlook

This thesis extends the existing theoretical and empirical literature on speculative stock-price bubbles as follows. First, based on the concept on rational expectations, an economic model is provided enabling the econometrician to directly estimate the specifications of nonlinear rational bubbles from stock-price data. The framework is established in a nonlinear state-space form where the bubble component is interpreted as the latent variable. For parameter estimation and estimation of the latent bubble component particle-filter methods are used. Second, this thesis analyzes stock-price volatility within the present-value framework when including rational bubbles. Third, this thesis establishes an alternative parametric specification for rational bubbles that is closely related to economic theory and to financial data.

Chapter 3 presents the economic nonlinear state-space model consisting of a stock's fundamental value and a latent bubble component described by the well-known periodically collapsing bubble specification proposed by Evans (1991). The model is not log-linearized and provides a clear-cut relationship between the stock-price, its fundamental value and the bubble that follows directly from the standard present-value model. A simulation study reveals a parameter identification problem, but demonstrates that the proposed estimation procedure is capable of identifying

the nonlinear state-space specification and the latent bubble process. The econometric procedure is applied to a real-world data set consisting of real stock-prices and dividends for the DAX, the S&P 500, the NASDAQ and the HIS. For all indices the estimation results indicate the presence of rational bubbles, but there is only evidence in favor of an explicit Evans bubble for the HSI. However, a goodness-of-fit test reveals some misspecifications of the economic model. Although the model is consistent with asset-pricing theory, one reason may be the use of the Evans-bubble specification as this model fails in modeling a realistic collapsing behavior.

Chapter 4 analyzes stock-price volatility in the presence of periodically collapsing Evans bubbles. Based on a present-value stock-price model a closed-form volatility formula is derived establishing a link between the Evans bubble and stock-price volatility. In a simulation study the econometric procedure presented in Chapter 3 is used to demonstrate how to fit the volatility formula to stock-market data. The major findings are that (a) Evans bubbles entail excess stock-price volatility, and (b) the model produces stock-price volatility paths that are broadly consistent with empirically observed volatility structures in the run-up to financial crises and the burst of a bubble.

Chapter 5 establishes a periodic and stochastically deflating bubble model. This new specification of rational bubbles is more parsimonious than the Evans bubble, empirically more plausible and allows the bubble to deflate stochastically over more than one period. The new bubble specification is estimated by particle-filter methods within a state-space framework (as presented in Chapter 3) using artificial as well as real-word data. A simulation study demonstrates the empirical identifiability of all model parameters. The estimation results for the real-world stock-price indices explicitly indicate the presence of a periodic, stochastically deflating bubble

in the data. A goodness-of-fit test affirms a good distributional fit of the bubble specification to the data, but also reveals some specification shortcomings. Finally, the theoretical stock-price volatility in the presence of the new periodic, stochastically deflating bubble is analyzed and the results are found to be in line with the volatility results established in Chapter 4 under the Evans bubble.

Based on these results, there are some interesting directions for future research. Irrespective of the explicit nonlinear rational bubble specification, the empirical applications of the economic framework may be interpreted to indicate some misspecifications. An explanation might be the simply structured fundamental value. In order to gain a further improved model fit, it could be useful to modify this fundamental value. To accomplish this, the challenge will be a realistic modeling of the fundamental value that is (a) based on observable variables, and (b) consistent with the rational-expectation model.

A further line of future research is the application of the economic nonlinear state-space model with periodic, stochastically deflating bubbles to other markets, like currency and commodity markets. As long as there exists an economic model describing a theoretical link between the asset price, its fundamental value and the latent bubble component, the model can be represented in a nonlinear state-space form and the latent bubble process can be estimated by particle-filter methods.

Finally, the results obtained in this thesis can be used to extend some previous empirical work on rational bubbles. For instance the work by Brooks and Katsaris (2005), describing the dynamics of stock-market returns, can be extended by incorporating an estimated latent Evans-bubble process. It could also be interesting to evaluate the power of frequently used bubble tests, like the sequential unit root tests

in Phillips et al. (2011) and Homm and Breitung (2012), in view of the empirically more plausible new periodic and stochastically deflating bubble process.

References

Abreu, D. and M. K. Brunnermeier (2002). Synchronization risk and delayed arbitrage. *Journal of Financial Economics 66*(2), 341–360.

Abreu, D. and M. K. Brunnermeier (2003). Bubbles and crashes. *Econometrica 71*(1), 173–204.

Al-Anaswah, N. and B. Wilfling (2011). Identification of speculative bubbles using state-space models with Markov-switching. *Journal of Banking and Finance 35*(5), 1073–1086.

Allen, F. and D. Gale (2000). Bubbles and crises. *The Economic Journal 110*(460), 236–255.

Andrieu, C., A. Doucet, and R. Holenstein (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B 72*(3), 269–342.

Baker, M. and J. Wurgler (2007). Investor sentiment in the stock market. *The Journal of Economic Perspectives 21*(2), 129–151.

Barro, R. J. (2006). Rare disasters and asset markets in the twentieth century. *The Quarterly Journal of Economics 121*(3), 823–866.

Blanchard, O. J. (1979). Speculative bubbles, crashes and rational expectations. *Economics Letters 3*(4), 387–389.

Blanchard, O. J. and M. W. Watson (1982). Bubbles, rational expectations and financial markets. In P. Wachtel (Ed.), *Crises in the Economic and Financial Structure*, pp. 295–315. Lexington Books, Lexington.

Bohl, M. T. (2003). Periodically collapsing bubbles in the US stock market? *International Review of Economics and Finance 12*(3), 385–397.

Briers, M., A. Doucet, and S. Maskell (2010). Smoothing algorithms for state–space models. *Annals of the Institute of Statistical Mathematics 62*(1), 61–89.

Brooks, C. and A. Katsaris (2005). A three-regime model of speculative behaviour: modelling the evolution of the S&P 500 composite index. *The Economic Journal 115*(505), 767–797.

Brunnermeier, M. K. (2008). Bubbles. In S. N. Durlauf and L. E. Blume (Eds.), *The New Palgrave Dictionary of Economics*. Palgrave Macmillan.

Brunnermeier, M. K. and J. Morgan (2010). Clock games: theory and experiments. *Games and Economic Behavior 68*(2), 532–550.

Brunnermeier, M. K. and M. Oehmke (2013). Bubbles, financial crises, and systematic risk. *Handbook of the Economics of Finance 2*, 1221–1288.

Camerer, C. (1989). Bubbles and fads in asset prices. *Journal of Economic Surveys 3*(1), 3–41.

Campbell, J. Y., A. W. Lo, and A. C. MacKinlay (1997). *The Econometrics of Financial Markets*. Princeton University Press, Princeton.

Campbell, J. Y. and R. J. Shiller (1987). Cointegration and tests of present value models. *The Journal of Political Economy 95*(5), 1062–1088.

Campbell, J. Y. and R. J. Shiller (1988). The dividend-price ratio and expectations of future dividends and discount factors. *Review of Financial Studies 1*(3), 195–228.

Cochrane, J. H. (2005). *Asset Pricing*. Princeton University Press, Princeton.

Creal, D. (2012). A survey of sequential Monte Carlo methods for economics and finance. *Econometric Reviews 31*(3), 245–296.

Cuthbertson, K. and D. Nitzsche (2004). *Quantitative Financial Economics: Stocks, Bonds and Foreign Exchange*. Wiley, New York.

De Long, J. B., A. Shleifer, L. H. Summers, and R. J. Waldmann (1990). Noise trader risk in financial markets. *Journal of Political Economy 98*(4), 703–738.

Dempster, A., N. Laird, and D. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B 39*(1), 1–38.

Diba, B. T. and H. I. Grossman (1988a). Explosive rational bubbles in stock prices? *The American Economic Review 78*(3), 520–530.

Diba, B. T. and H. I. Grossman (1988b). The theory of rational bubbles in stock prices. *The Economic Journal 98*(392), 746–754.

Diebold, F. X., T. A. Gunther, and A. S. Tay (1998). Evaluating density forecasts, with applications to financial risk management. *International Economic Review 39*(4), 863–883.

Douc, R. and O. Cappé (2005). Comparison of resampling schemes for particle filtering. In *4th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 64–69.

Doucet, A., N. de Freitas, and N. J. Gordon (2001). *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York.

Doucet, A., S. Godsill, and C. Andrieu (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing 10*(3), 197–208.

Doucet, A. and V. B. Tadic (2003). Parameter estimation in general state-space models using particle methods. *Annals of the Institute of Statistical Mathematics 55*(2), 409–422.

Driffill, J. and M. Sola (1998). Intrinsic bubbles and regime-switching. *Journal of Monetary Economics 42*(2), 357–373.

Duan, J.-C. and A. Fulop (2009). Estimating the structural credit risk model when equity prices are contaminated by trading noises. *Journal of Econometrics 150*(2), 288–296.

Duan, J. C. and A. Fulop (2011). A stable estimator of the information matrix under EM for dependent data. *Statistics and Computing 21*(1), 83–91.

Elliott, G., C. Granger, and A. Timmermann (2006). *Handbook of Economic Forecasting*, Volume 1. Elsevier, Amsterdam.

Enders, W. and P. L. Siklos (2001). Cointegration and threshold adjustment. *Journal of Business and Economic Statistics 19*(2), 166–176.

Evans, G. W. (1991). Pitfalls in testing for explosive bubbles in asset prices. *The American Economic Review 81*(4), 922–930.

Fernández-Villaverde, J. and J. F. Rubio-Ramírez (2007). Estimating macroeconomic models: a likelihood approach. *The Review of Economic Studies 74*(4), 1059–1087.

Flood, R. P. and R. J. Hodrick (1986). Asset price volatility, bubbles, and process switching. *The Journal of Finance 41*(4), 831–842.

Flood, R. P. and R. J. Hodrick (1990). On testing for speculative bubbles. *The Journal of Economic Perspectives 4*(2), 85–101.

Froot, K. A. and M. Obstfeld (1991). Intrinsic bubbles: the case of stock prices. *The American Economic Review 81*(5), 1189–1214.

Fukuta, Y. (1998). A simple discrete-time approximation of continuous-time bubbles. *Journal of Economic Dynamics and Control 22*(6), 937–954.

Garber, P. M. (1990). Famous first bubbles. *The Journal of Economic Perspectives 4*(2), 35–54.

Godsill, S. J., A. Doucet, and M. West (2004). Monte Carlo smoothing for nonlinear time series. *Journal of American Statistical Association 99*(465), 156–168.

Gordon, M. J. (1959). Dividends, earnings, and stock prices. *The Review of Economics and Statistics 41*(2), 99–105.

Gordon, N. J., D. J. Salmond, and A. F. M. Smith (1993). A novel approach to nonlinear and non-Gaussian Bayesian state estimation. *IEE Proceedings. Part F: Radar and Signal Processing 140*(2), 107–113.

Greene, W. H. (2012). *Econometric Analysis*. Pearson Education, New Jersey.

Gürkaynak, R. S. (2008). Econometric tests of asset price bubbles: taking stock. *Journal of Economic Surveys 22*(1), 166–186.

Hall, S. G., Z. Psaradakis, and M. Sola (1999). Detecting periodically collapsing bubbles: a Markov-switching unit root test. *Journal of Applied Econometrics 14*(2), 143–154.

Hamilton, J. D. and C. H. Whiteman (1985). The observable implications of self-fulfilling expectations. *Journal of Monetary Economics 16*(3), 353–373.

Harrison, J. M. and D. M. Kreps (1978). Speculative investor behavior in a stock market with heterogeneous expectations. *The Quarterly Journal of Economics 92*(2), 323–336.

Homm, U. and J. Breitung (2012). Testing for speculative bubbles in stock markets: a comparison of alternative methods. *Journal of Financial Econometrics 10*(1), 198–231.

Hsu, S. (2013). *Financial Crises, 1929 to the Present*. Edward Elgar Publishing Ltd, Cheltenham.

Hu, X.-L., T. B. Schön, and L. Ljung (2008). A basic convergence result for particle filtering. *IEEE Transactions on Signal Processing 56*(4), 1337–1348.

Hürzeler, M. and H. R. Künsch (1998). Monte Carlo approximations for general state-space models. *Journal of Computational and Graphical Statistics 7*(2), 175–193.

Hürzeler, M. and H. R. Künsch (2001). Approximating and maximising the likelihood for a general state space model. In A. Doucet, N. de Freitas, and N. J. Gordon (Eds.), *Sequential Monte Carlo Methods in Practice*, pp. 159–175. Springer Verlag, New York.

Ikeda, S. and A. Shibata (1992). Fundamentals-dependent bubbles in stock prices. *Journal of Monetary Economics 30*(1), 143–168.

Ikeda, S. and A. Shibata (1995). Fundamentals uncertainty, bubbles, and exchange rate dynamics. *Journal of International Economics 38*(3-4), 199–222.

Kantas, N., A. Doucet, S. S. Singh, and J. M. Maciejowski (2009). An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. In *15th IFAC Symposium on System Identification*, Volume 15, pp. 774–785.

Kim, J. and D. S. Stoffer (2008). Fitting stochastic volatility models in the presence of irregular sampling via particle methods and the EM algorithm. *Journal of Time Series Analysis 29*(5), 811–833.

Kim, S., N. Shephard, and S. Chib (1998). Stochastic volatility: likelihood inference and comparison with ARCH models. *The Review of Economic Studies 65*(3), 361–393.

Kindleberger, C. P. and R. Z. Aliber (2005). *Manias, Panics and Crashes: A History of Financial Crises*. John Wiley and Sons, Hoboken.

LeRoy, S. F. and R. D. Porter (1981). The present-value relation: tests based on implied variance bounds. *Econometrica 49*(3), 555–574.

Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, Series B 44*(2), 226–233.

Lucas, R. E. (1978). Asset prices in an exchange economy. *Econometrica 46*(6), 1429–1445.

Ma, Y. and A. Kanas (2004). Intrinsic bubbles revisited: evidence from nonlinear cointegration and forecasting. *Journal of Forecasting 23*(4), 237–250.

Malik, S. and M. K. Pitt (2011). Particle filters for continuous likelihood evaluation and maximisation. *Journal of Econometrics 165*(2), 190–209.

McLachlan, G. and T. Krishnan (2007). *The EM Algorithm and Extensions*. John Wiley and Sons, Hoboken.

Newey, W. K. and K. D. West (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica 55*(3), 703–708.

Olsson, J., O. Cappé, R. Douc, and E. Moulines (2008). Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models. *Bernoulli 14*(1), 155–179.

Phillips, P. C., Y. Wu, and J. Yu (2011). Explosive behavior in the 1990s NASDAQ: when did exuberance escalate asset values? *International Economic Review 52*(1), 201–226.

Phillips, P. C. and J. Yu (2011). Dating the timeline of financial bubbles during the subprime crisis. *Quantitative Economics 2*(3), 455–491.

Pitt, M. K. (2002). Smooth particle filters for likelihood evaluation and maximisation. Unpublished Manuscript, Department of Economics, Warwick University.

Pitt, M. K., S. Malik, and A. Doucet (2014). Simulated likelihood inference for stochastic volatility models using continuous particle filtering. *Annals of the Institute of Statistical Mathematics 66*(3), 527–552.

Poyiadjis, G., A. Doucet, and S. S. Singh (2005). Maximum likelihood parameter estimation in general state-space models using particle methods. In *Proceedings of the American Statistical Association*.

Reinhart, C. M. and K. Rogoff (2009). *This Time is Different: Eight Centuries of Financial Folly*. Princeton University Press, Princeton.

Rosenblatt, M. (1952). Remarks on a multivariate transformation. *The Annals of Mathematical Statistics 23*(3), 470–472.

Rotermann, B. and B. Wilfling (2014). Periodically collapsing Evans bubbles and stock-price volatility. *Economics Letters 123*(3), 383–386.

Scheinkman, J. A. and W. Xiong (2003). Overconfidence and speculative bubbles. *Journal of Political Economy 111*(6), 1183–1220.

Schön, T. B., A. Wills, and B. Ninness (2011). System identification of nonlinear state-space models. *Automatica 47*(1), 39–49.

Shiller, R. J. (1981). Do stock prices move too much to be justified by subsequent changes in dividends? *The American Economic Review 71*(3), 421–436.

Shiller, R. J. (2000). *Irrational Exuberance*. Princeton University Press, Princeton.

Shleifer, A. and R. W. Vishny (1997). The limits of arbitrage. *The Journal of Finance 52*(1), 35–55.

Simon, D. (2006). *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley, New Jersey.

Smith, V. L., G. L. Suchanek, and A. W. Williams (1988). Bubbles, crashes, and endogenous expectations in experimental spot asset markets. *Econometrica 56*(5), 1119–1151.

Tay, A. and K. F. Wallis (2000). Density forecasting: a survey. *Journal of Forecasting 19*(4), 235–254.

Tirole, J. (1982). On the possibility of speculation under rational expectations. *Econometrica 50*(5), 1163–1181.

Tirole, J. (1985). Asset bubbles and overlapping generations. *Econometrica 53*(5), 1071–1100.

Van Norden, S. and R. Vigfusson (1998). Avoiding the pitfalls: can regime-switching tests reliably detect bubbles? *Studies in Nonlinear Dynamics and Econometrics 3*(1), 1–24.

Vissing-Jorgensen, A. (2004). Perspectives on behavioral finance: Does irrationality disappear with wealth? Evidence from expectations and actions. In *NBER Macroeconomics Annual 2003, Volume 18*, pp. 139–208. The MIT Press.

West, K. D. (1987). A specification test for speculative bubbles. *The Quarterly Journal of Economics 102*(3), 553–580.

West, K. D. (1988). Dividend innovations and stock price volatility. *Econometrica 56*(1), 37–61.

Wu, Y. (1995). Are there rational bubbles in foreign exchange markets? Evidence from an alternative test. *Journal of International Money and Finance 14*(1), 27–46.

Wu, Y. (1997). Rational bubbles in the stock market: accounting for the US stock-price volatility. *Economic Inquiry 35*(2), 309–319.

Xiong, W. (2013). Bubbles, crises, and heterogeneous beliefs. In J.-P. Fouque and J. A. Langsam (Eds.), *Handbook on Systemic Risk*, pp. 663–713. Cambridge University Press, Cambridge.

Appendix

Programming codes for Chapter 3

In the following we provide the MATLAB-Code for the identification and estimation of our economic nonlinear state-space model as introduced in Chapter 3. The main file main_chapter3.m starts with the programming code for the simulation study of the economic model with periodically collapsing Evans bubbles. After this, the file gives the general programming code for the particle based EM algorithm to identify the nonlinear state-space model. At the end, the main file includes the general programming codes for evaluating the estimation results, that is the estimation of the latent Evans-bubble process, the computation of the standard errors and the goodness-of-fit tests.

At first we give the programming code of the main file. After that, the other m-files needed are given in the order of their occurrence in the main file. The codes for the EM algorithm via particle-filter methods are based on the descriptions in Schön et al. (2011).

main_chapter3.m

```
%%%%% Main script - Chapter 3
%%%%% Simulate data for the nonlinear state-space model
% Number of data.
T = 250;
% Parameter setting for the nonlinear state-space model.
% Set the required rate of return.
r = 0.02;
```

```
% Set the parameter of the observation equation.
m.phi = 50;
m.sig2 = 1.2;
% Set the parameter of the Evans model (state equation).
m.psi = 1/(1+r);
m.iota2 = 0.001;
m.kappa = 1.1;
m.p = 0.98;
m.tau = 2;

% Dividend process (driftless random walk with sd 0.03).
D = zeros(1,T); noise = randn(1,T);
% Set the starting value.
D(1)=0.3;
% Simulate data of the dividend process.
for t=1:(T-1)
  D(t+1) = D(t) + 0.03*noise(t);
end;

% Evans-bubble process.
Bt = zeros(1,T); y = randn(1,T); nu = binornd(1,m.p,1,T);
% Set the starting value.
Bt(1) = 0.5;
% Simulate data of the Evans bubble.
for t=1:T-1
  if Bt(t) <= m.tau
    Bt(t+1) = (1/m.psi)*Bt(t)*exp(sqrt(m.iota2)*y(t)-(0.5*m.iota2));
  else
    Bt(t+1) = (m.kappa +(1/(m.p*m.psi))*(Bt(t)-(m.kappa*m.psi)).*...
                    nu(t)).*exp(sqrt(m.iota2)*y(t)-(0.5*m.iota2));
  end;
end;
```

```
% Calculate the stock-price process.
varepsilon = randn(1,T);
P = m.phi*D + Bt + sqrt(m.sig2)*varepsilon;

% Save the data in the structure array z.
z.P = P;
z.D = D;
z.BTrue = Bt(1:T);

%%%%% Parameter estimation
%%%%% EM algorithm using the particle filter and particle smoother
% We need the data set given in the structure array z. For the
% artificial and real-world data sets, read the (real) prices and
% (real) dividends from the corresponding Excel file and save the
% prices as z.P and dividends as z.D in the structure array z.

% Set max. number of iterations, if no convergence criterion is used!
% opt.maximumiter = 500;
% Set the number of particles for the EM algorithm.
N  = 300;
% Set the parameter vector theta_0 to initiate the EM algorithm.
phi0 = 20;
sig20 = 0.5;
psi0 = 0.8;
iota20 = 0.005;
kappa0 = 0.5;
p0 = 0.8;
theta0 = [phi0 sqrt(sig20) psi0 sqrt(iota20) kappa0 p0];
% Define the vector for the parameter estimates.
mEst = m;
mEst.phi = theta0(1);
mEst.sig2 = theta0(2)^2;
mEst.psi = theta0(3);
```

```
mEst.iota2 = theta0(4)^2;
mEst.kappa = theta0(5);
mEst.p = theta0(6);
% Owing to an identification problem, we refrained from estimating
% the parameter tau, so we set tau fixed.
mEst.tau = 2; % !Change tau here!

% Store the iterates.
theta = theta0;
theta_k(1,1:6) = theta0;
mEst_k(1)= mEst;
crit_value(1) = 1;

% Set constraints on the parameter vector.
lb = [0.000001,0.000001,0.5,0.000001,0.000001,0.000001];
ub = [250,8,0.999999,5,mEst.tau,0.999999];
% Define some optimization conditions.
options = optimset('Algorithm','active-set','Display','off',...
                   'TolFun',1e-6,'TolX',1e-5);
% Fix the random generator to start each iteration with the same
% random numbers. This helps to minimize some random effects of
% the Monte Carlo approach.
stream = RandStream.getGlobalStream;
% General Setting: savedState = stream.State;
% To reproduce the results use this random number stream.
savedState = uint32(xlsread('savedState'));
% Use loop if no termination condition is used!
% for k = 1:opt.maximumiter
% Or define the convergence criterion.
c = 1/(N);
```

```
%%% Start of the EM algorithm.
k = 2;
while (crit_value(k-1) > c)
  % Set the random stream in each iteration k.
  stream.State = savedState;
  % Run the particle filter and particle smoother to obtain
  % particles from the sequential importance sampling and the
  % corresponding smoothed weights.
  %%% Expectation step at iteration k.
  % Run the particle filter.
  PF = particle_filter3(mEst,N,z);
  % Run the particle smoother.
  PS = particle_smoother3(mEst,N,PF);
  % Calculate the Q-function subject to theta_k.
  Q_value_k_k = -(Q3_theta_k(theta,mEst,PF.xPWeighted,PS.wij,...
                  PS.wT,z.P,z.D));
  %%% Maximization step at iteration k.
  % Maximize the Q-function subject to theta and obtain theta_k+1.
  [theta,Qval] = fmincon('Q3_theta_k',theta,[],[],[],[],lb,ub,[],...
                   options,m.Est,PF.xPWeighted,PS.wij,PS.wT,z.P,z.D);
  % Calculate the Q-function subject to theta_k+1.
  Q_value_k_k1 = -Qval;
  % Store the parameter estimates in iteration k.
  mEst.phi = theta(1);
  mEst.sig2 = theta(2)^2;
  mEst.psi = theta(3);
  mEst.iota2 = theta(4)^2;
  mEst.kappa = theta(5);
  mEst.p = theta(6);
  % Store the iterates.
  theta_k(k,1:6)= theta;
  mEst_k(k) = mEst;
  % Compute critical value to check the convergence criterion.
```

```
  crit_value(k) = Q_value_k_k1-Q_value_k_k;
  % Print the parameter estimates.
  disp(['Iteration nr: ' num2str((k-1))...
       ' Estimates, phi: ' num2str(mEst.phi),...
       ' sig2: ',num2str(mEst.sig2),'  psi: ',num2str(mEst.psi),...
       ' iota2: ',num2str(mEst.iota2),...
       '  kappa: ',num2str(mEst.kappa),...
       ' p: ',num2str(mEst.p),' tau: ',num2str(mEst.tau)])
  % If crit_value(k) > c: k is updated to k+1 and a new iteration
  % is conducted.
  k = k+1;
end

%%%%% Evaluating the estimation results
% Fix the random generator to reproduce the results.
stream = RandStream.getGlobalStream;
savedState = uint32(xlsread('savedState'));
stream.State = savedState;

%%% Estimation of the latent Evans-bubble process
% Set number of particles.
N=500;
% Use the estimated parameter set.
mEst = mEst_k(k-1);
% Compute the particles and smoothed weights.
% Run the particle filter.
PF = particle_filter3(mEst,N,z);
% Run the particle smoother.
PS = particle_smoother3(mEst,N,PF);
% Compute the smoothed states.
for t=1:T
  xs(t) = sum(PS.wT(t,:).*PF.xPWeighted(t,:));
end;
```

```
%%% Standard errors
% Compute the standard errors of the parameter estimates by the use
% of the stable estimator of the information matrix established by
% Duan and Fulop (2011).
standard_error3(PF.xPWeighted,PS.wT,PS.wij,z.P,z.D,mEst,N)

%%% Goodness-of-fit test
% Compute the PITs and the corresponding empirical cdf.
pit = forecast_density(PF,mEst,z);
[cdf_pit]=ecdf(pit);

% KS-test.
% Simulate 1 million uniforms.
r_unif = unifrnd(0,1,1000000,1);
% Compare these uniforms with the PITs by using a KS-Test.
[h,pValue,ks_stat] = kstest2(r_unif,pit)

% LB-test.
[h,pValue,lb_stat,cValue] = lbqtest(pit)
```

particle_filter3.m

```
%%% Particle filter
function pf = particle_filter3(m,N,z)
  P = z.P;
  D = z.D;
  T = size(P,2);
  xf = zeros(1,size(P,2));
  xPWeighted = zeros(T,N);
  xPResampled = zeros(T,N);
  wt = zeros(T,N);
  % 1. Initialize particles.
  % Initial state unknown, assumed to be a positive random variable.
    B = exp(sqrt(0.2)*randn(1,N)-(2*0.2));
  % 2. Run the particle filter.
  for t=1:T
    % Compute the weights.
    w = exp((-0.5/m.sig2)*((repmat(P(t),1,N) -...
            m.phi.*(repmat(D(t),1,N)) - B).^2));
    w(w==0) = exp(-745);
    w = w/sum(w);
    % Save the particles and corresponding weights.
    xPWeighted(t,:) = B;
    wt(t,:) = w;
    % Compute state estimate, see Eq. (3.13).
    xf(t) = sum(w.*B);
    % Resample the particles.
    index = sysresample(w);
    B = B(index);
    xPResampled(t,:) = B;
    % Sequential importance sampling, which produces
    % new particles for t+1.
    idx1 = find(B <= m.tau);
```

```
    idx2 = find(B > m.tau);
    B(idx1) = (1/m.psi)*B(idx1).*exp(sqrt(m.iota2)*...
              randn(1,length(idx1))-(0.5*m.iota2));
    B(idx2) = (m.kappa+(1/((m.p)*m.psi))*(B(idx2)-(m.kappa*...
              m.psi)).*binornd(1,(m.p),1,length(idx2))).*...
              exp(sqrt(m.iota2)*randn(1,length(idx2))-...
              (0.5*m.iota2));
  end;
  pf.Xf = xf;
  pf.xPWeighted = xPWeighted;
  pf.xPResampled = xPResampled;
  pf.w = wt;
end
```

sysresample.m

```
% Systematic Resampling of the particles.
function i=sysresample(q)
 qc = cumsum(q);
 M = length(q);
 u = ([0:M-1]+rand(1))/M;
 i = zeros(1,M);
 k = 1;
   for j = 1:M
     while (qc(k)<u(j))
       k = k+1;
     end
     i(j) = k;
   end
end
```

particle_smoother3.m

```
%%% Particle smoother
function ps = particle_smoother3(m,N,PF)
  T = size(PF.xPWeighted,1);
  wT = zeros(T,N);
  kk = ones(1,N);

  % 1. Set the smoothed weights in T.
  wT(T,:) = PF.w(T,:);
  % 2. Run the particle smoother.
  for t = T-1:-1:1
    p_xt1_xt = zeros(N,N);
    % Use the computed particles in t and t+1 form
    % sequential importance sampling.
    xt = PF.xPWeighted(t,:);
    xt1 = PF.xPWeighted(t+1,:);
    idx1 = find(xt <= m.tau);
    idx2 = find(xt >  m.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
    % Compute the density of all possible particle combinations.
    p_xt1_xt(idx2,:) = (m.p)*lognpdf(repmat(xt1,length(idx2),1),...
                       (-(m.iota2/2) + log(m.kappa+(1/(m.p*m.psi))*...
                       (xti2-m.kappa*m.psi))),sqrt(m.iota2))+...
                       (1-(m.p))*lognpdf(repmat(xt1,length(idx2),1),...
                       (-(m.iota2/2) + log(m.kappa)),sqrt(m.iota2));
    p_xt1_xt(idx1,:) = lognpdf(repmat(xt1,length(idx1),1),...
                       (-(m.iota2/2)+log((1/m.psi)*xti1)),sqrt(m.iota2));
    % Compute the (normalized) smoothed weights in Eq.(3.20).
    % 1. Compute the denominator of the summand in Eq. (3.20).
    vk = PF.w(t,:)*p_xt1_xt;
    % Set all weighted densities which are approx. zero on minimal
```

```
    % positive value to ensure numerical stability in the next step.
    vk(vk==0) = exp(-745);
    % 2. Compute the complete sum in Eq. (3.20).
    wij(:,:,t) = (repmat(PF.w(t,:),N,1)'.*repmat(wT(t+1,:),N,1).*...
                  p_xt1_xt)./repmat(vk,N,1);
    k = (repmat(wT(t+1,:),N,1).*p_xt1_xt)./repmat(vk,N,1);
    sk = k*kk';
    % 3. Compute Eq.(3.20).
    wT(t,:) = PF.w(t,:).*sk';
  end;
  ps.wT = wT;
  ps.wij = wij;
end
```

Q3_theta_k.m

```
function Qfun = Q3_theta_k(theta,mEst,xPWeighted,wij,wT,P,D)
  % Parameter for maximizing.
  phi = theta(1);
  sig2 = theta(2);
  psi = theta(3);
  iota2 = theta(4);
  kappa = theta(5);
  p = theta(6);
  % Sample size.
  T = size(xPWeighted,1);
  % Number of particles.
  N = size(xPWeighted,2);
  Qfun3 = 0;
  Qfun2 = 0;
  kk = ones(1,N);
```

```
  % Compute the second term of the approximated Q-function given in
  % Eq.(3.30).
  for t=1:T-1
    p_xt1_xt_k = zeros(N,N);
    xt = xPWeighted(t,:);
    xt1 = xPWeighted(t+1,:);
    idx1 = find(xt <= mEst.tau);
    idx2 = find(xt > mEst.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
    p_xt1_xt_k(idx2,:) = (p)*lognpdf(repmat(xt1,length(idx2),1),...
                         (-(iota2^2/2)+log(kappa+(1/(p*psi))*...
                         (xti2-kappa*psi))),sqrt(iota2^2))+...
                         (1-(p))*lognpdf(repmat(xt1,length(idx2),1),...
                         (-(iota2^2/2) + log(kappa)),sqrt(iota2^2));
    p_xt1_xt_k(idx1,:) = lognpdf(repmat(xt1,length(idx1),1),...
                       (-(iota2^2/2)+log((1/psi)*xti1)),sqrt(iota2^2));
    log_p_xt1_xt_k = log(p_xt1_xt_k);
    % To ensure numerical stability (20 times the computer tolerance).
    log_p_xt1_xt_k(log_p_xt1_xt_k == -Inf)= -15000;
    wij_p = sum((wij(:,:,t).*(log_p_xt1_xt_k))*kk');
    Qfun2 = Qfun2 + wij_p;
  end;
  % Compute the third term of the approximated Q-function given in
  % Eq.(3.30).
  for t=1:T
    p_yt_xt = -(1/2)*log(sig2^2)-(1/(2*sig2^2))*(P(t)-phi*D(t)-...
              xPWeighted(t,:)).^2 ;
    Qfun3 = Qfun3 + wT(t,:)*p_yt_xt';
  end;
  Qfun = Qfun3 + Qfun2 ;% Qfun1 is neglected.
  Qfun = -Qfun;
end
```

standard_error3.m

```
function SE = standard_error3(xPWeighted,wT,wij,P,D,m,N)
  T = size(xPWeighted,1);
  kk = ones(1,N);
  f11 = zeros(6,T);

  % Compute the second term of the score of the observed-data
  % log-likelihood function given in Eq. (3.40).
  for t = 2:T
    xt = xPWeighted(t-1,:);
    xt1 = xPWeighted(t,:);
    idx1 = find(xt <= m.tau);
    idx2 = find(xt > m.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
    f3a = zeros(N,N);
    f4a = zeros(N,N);
    f5a = zeros(N,N);
    f6a = zeros(N,N);
    f1 = zeros(6,1);
    % Compute the first derivatives of the joint log-likelihood
    % function with respect to theta for all N paths. We compute these
    % first derivatives with the mathematical software "Maple".
    % First derivatives with respect to phi and sig2 do not dependent
    % on the latent bubble (particles).
    % First derivative with respect to phi.
    f1a = (P(t)-m.phi.*D(t)-xt1).*D(t)./m.sig2 ;
    % First derivative with respect to sig2.
    f2a = -0.5.*(-m.phi.^2.*D(t).^2+2.*D(t).*((-xt1)+P(t)).*m.phi-...
          P(t).^2-(xt1).^2+2.*P(t).*(xt1)+m.sig2)./(sqrt(m.sig2).*...
          sqrt(m.sig2.^3));
    % Case 1: particles are smaller than tau.
```

```
% First derivative with respect to psi.
f3a(idx1,:) = -(log(repmat(xt1,length(idx1),1))+(1./2).*m.iota2-...
              log(xti1./m.psi))./(m.iota2.*m.psi);
% First derivative with respect to iota2
f4a(idx1,:) = (((0.5).*(-(0.5).*(log(repmat(xt1,length(idx1),1))+...
            (1./2).*m.iota2-log(xti1./m.psi))./m.iota2+(1./2).*...
            (log(repmat(xt1,length(idx1),1))+(1./2).*m.iota2-...
            log(xti1./m.psi)).^2./m.iota2.^2).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx1),1))+(1./2).*m.iota2-...
            log(xti1./m.psi)).^2./m.iota2).*sqrt(2)./sqrt(pi.*...
            m.iota2.*repmat(xt1,length(idx1),1).^2)-(1./4).*...
            exp(-(1./2).*(log(repmat(xt1,length(idx1),1))+(1./2).*...
            m.iota2-log(xti1./m.psi)).^2./m.iota2).*sqrt(2).*pi.*...
            repmat(xt1,length(idx1),1).^2./(pi.*m.iota2.*...
            repmat(xt1,length(idx1),1).^2).^(3./2)).*sqrt(2).*...
            sqrt(pi.*m.iota2.*repmat(xt1,length(idx1),1).^2))./...
            (exp(-(1./2).*(log(repmat(xt1,length(idx1),1))+...
            (1./2).*m.iota2-log(xti1./m.psi)).^2./m.iota2));
% First derivative with respect to kappa.
f5a(idx1,:) = 0;
% First derivative with respect to p.
f6a(idx1,:) = 0;
% Case 2: particles are bigger than tau.
% First derivative with respect to psi.
f3a(idx2,:) = (1./(m.p.*exp(-(log(repmat(xt1,length(idx2),1))+.....
            (1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
            (m.psi.*m.p))).^2./(2.*m.iota2))./sqrt(2.*pi.*...
            m.iota2.*repmat(xt1,length(idx2),1).^2)+(1-m.p).*...
            exp(-(log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa)).^2./(2.*m.iota2))./sqrt(2.*pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2))).*(0.5.*(m.p.*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).*...
```

```
            (-(xti2-m.kappa.*m.psi)./(m.psi.^2.*m.p)-m.kappa./...
            (m.psi.*m.p)).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).^2./...
            m.iota2).*sqrt(2))./(m.iota2.*(m.kappa+(xti2-m.kappa.*...
            m.psi)./(m.psi.*m.p)).*sqrt(pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2)));
% First derivative with respect to iota2.
f4a(idx2,:) = (1./(m.p.*exp(-(log(repmat(xt1,length(idx2),1))+...
            (1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
            (m.psi.*m.p))).^2./(2.*m.iota2))./sqrt(2.*pi.*...
            m.iota2.*repmat(xt1,length(idx2),1).^2)+(1-m.p).*...
            exp(-(log(repmat(xt1,length(idx2),1))+(0.5).*...
            m.iota2-log(m.kappa)).^2./(2.*m.iota2))./sqrt(2.*...
            pi.*m.iota2.*repmat(xt1,length(idx2),1).^2))).*...
            (0.5.*(m.p.*(-(0.5).*(log(repmat(xt1,length(idx2)...
            ,1))+(1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*...
            m.psi)./(m.psi.*m.p)))./m.iota2+(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./...
            2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
            (m.psi.*m.p))).^2./m.iota2.^2).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*...
            m.p))).^2./m.iota2).*sqrt(2))./(sqrt(pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2))-0.25.*(m.p.*...
            exp(-(1./2).*(log(repmat(xt1,length(idx2),1))+...
            (1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
            (m.psi.*m.p))).^2./m.iota2).*sqrt(2).*pi.*...
            repmat(xt1,length(idx2),1).^2)./((pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2).^(3./2))+0.5.*((1-...
            m.p).*(-(1./2).*(log(repmat(xt1,length(idx2),1))+...
            (1./2).*m.iota2-log(m.kappa))./m.iota2+(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
```

```
            log(m.kappa)).^2./m.iota2.^2).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa)).^2./m.iota2).*sqrt(2))./(sqrt(pi.*...
            m.iota2.*repmat(xt1,length(idx2),1).^2))-0.25.*...
            ((1-m.p).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*...
            m.iota2-log(m.kappa)).^2./m.iota2).*sqrt(2).*pi.*...
            repmat(xt1,length(idx2),1).^2)./((pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2).^(3./2)));
% First derivative with respect to kappa.
f5a(idx2,:) = (1./(m.p.*exp(-(log(repmat(xt1,length(idx2),1))+...
            (1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
            (m.psi.*m.p))).^2./(2.*m.iota2))./sqrt(2.*pi.*...
            m.iota2.*repmat(xt1,length(idx2),1).^2)+(1-m.p).*exp(-...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa)).^2./(2.*m.iota2))./sqrt(2.*pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2))).*(0.5.*(m.p.*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).*...
            (1-1./m.p).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).^2./...
            m.iota2).*sqrt(2))./(m.iota2.*(m.kappa+(xti2-m.kappa.*...
            m.psi)./(m.psi.*m.p)).*sqrt(pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2))+0.5.*((1-m.p).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa)).*exp(-(1./2).*...
            (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
            log(m.kappa)).^2./m.iota2).*sqrt(2))./(m.iota2.*...
            m.kappa.*sqrt(pi.*m.iota2.*...
            repmat(xt1,length(idx2),1).^2)));
% First derivative with respect to p.
f6a(idx2,:) = (1./(m.p.*exp(-(log(repmat(xt1,length(idx2),1))+...
```

```
                (1./2).*m.iota2-log(m.kappa+(xti2-m.kappa.*m.psi)./...
                (m.psi.*m.p))).^2./(2.*m.iota2))./sqrt(2.*pi.*...
                m.iota2.*repmat(xt1,length(idx2),1).^2)+(1-m.p).*...
                exp(-(log(repmat(xt1,length(idx2),1))+(0.5).*m.iota2-...
                log(m.kappa)).^2./(2.*m.iota2))./sqrt(2.*pi.*m.iota2.*...
                repmat(xt1,length(idx2),1).^2))).*(0.5.*(exp(-(1./2).*...
                (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
                log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).^2./...
                m.iota2).*sqrt(2))./(sqrt(pi.*m.iota2.*...
                repmat(xt1,length(idx2),1).^2))-0.5.*...
                ((log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
                log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).*...
                (xti2-m.kappa.*m.psi).*exp(-(1./2).*...
                (log(repmat(xt1,length(idx2),1))+(1./2).*m.iota2-...
                log(m.kappa+(xti2-m.kappa.*m.psi)./(m.psi.*m.p))).^2./...
                m.iota2).*sqrt(2))./(m.p.*m.iota2.*m.psi.*(m.kappa+...
                (xti2-m.kappa.*m.psi)./(m.psi.*m.p)).*sqrt(pi.*...
                m.iota2.*repmat(xt1,length(idx2),1).^2))-0.5.*...
                (exp(-(1./2).*(log(repmat(xt1,length(idx2),1))+(1./...
                2).*m.iota2-log(m.kappa)).^2./m.iota2).*sqrt(2))./...
                (sqrt(pi.*m.iota2.*repmat(xt1,length(idx2),1).^2)));
  % Compute the weighted sum of the derivatives (expectated value).
  f1(1) = wT(t,:)*f1a';
  f1(2) = wT(t,:)*f2a';
  f1(3) = sum((wij(:,:,t-1).*(f3a))*kk') ;
  f1(4) = sum((wij(:,:,t-1).*(f4a))*kk') ;
  f1(5) = sum((wij(:,:,t-1).*(f5a))*kk') ;
  f1(6) = sum((wij(:,:,t-1).*(f6a))*kk') ;
  f1(isnan(f1))=0; % To ensure numerical stability.
  % Sum over t = 2, ..., T.
  f11(:,t)=f11(:,t)+(f1);
end
```

```
% First derivatives of the joint log-likelihood function at t=1.
xt = xPWeighted(1,:);
f1 = zeros(6,1);
% First derivative with respect to phi.
f1a = (P(1)-m.phi.*D(1)-xt).*D(1)./m.sig2 ;
% First derivative with respect to sig2.
f2a = -0.5.*(-m.phi.^2.*D(1).^2+2.*D(1).*(-xt+P(1)).*m.phi-P(1).^2-...
       xt.^2+2.*P(1).*xt+m.sig2)./(sqrt(m.sig2).*sqrt(m.sig2.^3));
% First derivative with respect to psi.
%f3a = 0;
% First derivative with respect to iota2.
%f4a = 0);
% First derivative with respect to kappa.
%f5a = 0;
% First derivative with respect to p.
%f6a = 0;
% Compute the weighted sum of the first derivatives.
f1(1) = wT(1,:)*f1a';
f1(2) = wT(1,:)*f2a';
f1(3) = 0;
f1(4) = 0;
f1(5) = 0;
f1(6) = 0;
f11(:,1) = (f1);
% Expected value of the derivative of the joint log-likelihood.
score = f11;
% Take into account the dependence among the lagged terms by the
% procedure of Newey and West (1987).
A = 0;
TT = size(score,2);
```

```
  % Compute the several covariance terms Aj given Eq. (3.42).
  for j = 0:(TT-1)
    A = 0;
    for t = 1:(TT-(j))
      A1 = score(:,t)*score(:,t+j)';
      A = A+A1;
    end
    Aj(:,:,(j+1)) = A;
  end
  format long
  % Compute the negative expected value of the Hessian, see Eq. (3.41).
  hessian = Aj(:,:,1);
  lag = 15;
  for j = 1:lag
    hessian = hessian+(1-j./(lag+1)).*(Aj(:,:,(j+1))+Aj(:,:,(j+1))');
  end
  % Compute the covariance matrix of the parameter estimates by taking
  % the inverse of the hessian.
  Cov = inv(hessian);
  % Standard errors are given by the diagonal of the covariance matrix.
  Cov_d = diag(Cov);
  % Results.
  SE.phi = Cov_d(1);
  SE.sig2 = Cov_d(2);
  SE.psi = Cov_d(3);
  SE.iota2 = Cov_d(4);
  SE.kappa = Cov_d(5);
  SE.p = Cov_d(6);
end
```

forecast_density.m

```
function PIT = forecast_density(PF,mEst,z)
  N = size(PF.xPWeighted,2);
  T = size(PF.xPWeighted,1);
  PIT = zeros(T,1);
  P  = z.P;
  D = z.D;
  % Compute the PIT's by Eq. (3.47) and Eq. (3.49).
  for t=1:T
    density = normcdf(repmat(P(t),1,N),mEst.phi.*(repmat(D(t),1,N))+...
                      PF.xPWeighted(t,:),sqrt(mEst.sig2));
    PIT(t)=sum(density)/N;
  end;
end
```

Programming codes for Chapter 4

In the following we provide the MATLAB-Code for the identification of our economic nonlinear state-space model and the estimation of the latent Evans bubble using the particle-filter methods. The main file main_chapter4.m contains the programming code for the simulation study as well as the general programming code for the particle based EM algorithm to identify the nonlinear state-space model. At the end, this file includes the general programming codes for the the estimation of the latent Evans-bubble process, the computation of the standard errors and the computation of the conditional stock-price volatility path.

At first we give the programming code of the main file. After that, the other m-files needed are given in the order of their occurrence in the main file. The codes for the EM algorithm via particle-filter methods are based on the descriptions in Schön et al. (2011).

main_chapter4.m

```
%%%%% Main script - Chapter 4
%%%%% Simulate data for the nonlinear state-space model

% Number of data.
T = 250;
% Parameter setting for the nonlinear state-space model.
% Set the model parameters.
m.r = 0.02;
m.sig2 = 0.03^2;
m.iota2  = 0.001;
m.kappa  = 1.1;
m.p = 0.98;
m.tau = 2;
```

```
% Dividend process (driftless random walk).
D = zeros(1,T+1); epsilon = randn(1,T+1);
% Set the starting value.
D(1)=0.3;
% Simulate dividend process form t-1,t, ...,N-1 => 250 observations
for t=2:(T+1)
  D(t) = D(t-1) + sqrt(m.sig2)*epsilon(t);
end;

% Evans-bubble process.
Bt = zeros(1,T); y = randn(1,T); nu = binornd(1,m.p,1,T);
% Set the starting value.
Bt(1) = 0.5;
% Simulate data of the Evans bubble.
for t=1:T-1
  if Bt(t) <= m.tau
    Bt(t+1) = (1/(1/(1+m.r)))*Bt(t)*exp(sqrt(m.iota2)*y(t)-...
              (0.5*m.iota2));
  else
    Bt(t+1) = (m.kappa +(1/(m.p*(1/(1+m.r))))*(Bt(t)-(m.kappa*(1/(1+...
               m.r)))).*nu(t)).*exp(sqrt(m.iota2)*y(t)-(0.5*m.iota2));
  end;
end;

% Calculate the stock-price process, given by Eq.(4.3) and Eq.(4.4).
P   =(1/m.r)*D(2:(T+1)) + Bt;

% Save the data in the structure array z.
z.P = P;
z.D = D(1:(T));
z.BTrue = Bt;
```

```
%%%%% Parameter estimation
%%%%% EM algorithm using the particle filter and particle smoother
% We need the data set given in the structure array z.
% Set the number of particles for the EM algorithm.
N  = 300; % Number of particles
% Set the parameter vector theta_0 to initiate the EM algorithm.
r0 = 0.03;
sig20 = 0.5;
iota20 = 0.005;
kappa0 = 0.5;
p0 = 0.8;
theta0 = [r0 sqrt(sig20) sqrt(iota20) kappa0 p0];
% Define the vector for the parameter estimates.
mEst = m;
mEst.r = theta0(1);
mEst.sig2 = theta0(2)^2;
mEst.iota2 = theta0(3)^2;
mEst.kappa = theta0(4);
mEst.p = theta0(5);
% Owing to an identification problem, we refrained from estimating
% the parameter tau and use tau=2 instead.
mEst.tau = 2;

% Store the iterates.
theta = theta0;
theta_k(1,1:5) = theta0;
mEst_k(1)= mEst;
crit_value(1) = 1;

% Set constraints on the parameter vector.
lb = [0.000001,0.000001,0.000001,0.000001,0.000001];
ub = [1,8,5,mEst.tau,0.999999];
% Define some optimization conditions.
```

```
options = optimset('Algorithm','active-set','Display','off',...
                   'TolFun',1e-6,'TolX',1e-5);

% Fix the random generator to start each iteration with the same
% random numbers. This helps to minimize some random effects of this
% Monte Carlo approach.
stream = RandStream.getGlobalStream;
savedState = stream.State;
% Define the convergence criterion.
c = 1/(N);

%%% Start of the EM algorithm.
k = 2;
while (crit_value(k-1) > c)
  % Set the random stream in each iteration k.
  stream.State = savedState;
  % Run the particle filter and particle smoother to obtain particles
  % from the sequential importance sampling and the corresponding
  % smoothed weights.
  %%% Expectation step at iteration k.
  % Run the particle filter.
  PF = particle_filter4(mEst,N,z);
  % Run the particle smoother.
  PS = particle_smoother4(mEst,N,PF);
  % Calculate the Q-function subject to theta_k.
  Q_value_k_k = -(Q4_theta_k(theta,mEst,PF.xPWeighted,PS.wij,...
                 PS.wT,z.P,z.D));
  %%% Maximization step at iteration k.
  % Maximize the Q-function subject to theta and obtain theta_k+1.
  [theta,Qval] = fmincon('Q4_theta_k',theta,[],[],[],[],lb,ub,[],...
                   options,mEst,PF.xPWeighted,PS.wij,PS.wT,z.P,z.D);
  % Calculate the Q-function subject to theta_k+1.
  Q_value_k_k1 = -Qval;
```

```
  % Store the parameter estimates in iteration k.
  mEst.r = theta(1);
  mEst.sig2 = theta(2)^2;
  mEst.iota2 = theta(3)^2;
  mEst.kappa = theta(4);
  mEst.p = theta(5);
  % Store the iterates.
  theta_k(k,1:5) = theta;
  mEst_k(k) = mEst;
  % Compute critical value to check the convergence criterion.
  crit_value(k) = Q_value_k_k1-Q_value_k_k;
  % Print the parameter estimates.
  disp(['Iteration nr: ' num2str((k-1))...
        ' Estimates, r: ' num2str(mEst.r),...
        ' sig2: ',num2str(mEst.sig2),' iota2: ',num2str(mEst.iota2),...
        ' kappa: ',num2str(mEst.kappa),' p: ',num2str(mEst.p),...
        ' tau: ',num2str(mEst.tau)])
  % If crit_value(k) > c: k is updated to k+1 and a new iteration is
  % conducted.
  k = k+1;
end

%%%%% Evaluating the estimation results
% Fix the random generator to reproduce the results.
stream = RandStream.getGlobalStream;
savedState = uint32(xlsread('savedState'));
stream.State = savedState;

%%% Estimation of the latent Evans-bubble Process
% Set number of particles.
N=500;
% Use the estimated parameter set.
mEst = mEst_k(k-1);
```

```
% Compute the particles and smoothed weights
% Run the particle filter.
PF = particle_filter4(mEst,N,z);
% Run the particle smoother.
PS = particle_smoother4(mEst,N,PF);
% Compute the smoothed states.
for t=1:T
  xs(t) = sum(PS.wT(t,:).*PF.xPWeighted(t,:));
end;

%%% Standard errors
% Compute the standard errors of the parameter estimates by the use
% of the stable estimator of the information matrix established by
% Duan and Fulop (2011).
standard_error4(PF.xPWeighted,PS.wT,PS.wij,z.P,z.D,mEst,N)

%%% Conditional stock-price variance with Evans bubble
% Compute the conditional variance given in Eq. (4.10).
beta = 1/mEst.r; psi = (1/(1+mEst.r));
Var_t1 = zeros(1,T);
for t=2:T
  if xs(t-1) <= mEst.tau
    Var_t1(t) = (((1/psi)*xs(t-1))^2*(exp(mEst.iota2)-1)+...
                 beta^2*mEst.sig2);
  else
    Var_t1(t) = ((mEst.kappa^2+2*mEst.kappa*((xs(t-1)-mEst.kappa*...
                psi)/psi))*(exp(mEst.iota2)-1)+((xs(t-1)-mEst.kappa*...
                psi)/(mEst.p*psi))^2*(exp(mEst.iota2)*mEst.p-...
                mEst.p^2)+beta^2*mEst.sig2);
  end;
end;
```

particle_filter4.m

```
%%% Particle filter
function pf = particle_filter4(m,N,z)
  P = z.P;
  D = z.D;
  T = size(P,2);
  xf = zeros(1,size(P,2));
  xPWeighted = zeros(T,N);
  xPResampled = zeros(T,N);
  wt = zeros(T,N);

  % 1. Initialize particles.
  % Initial state unknown, assumed to be a positive random variable.
    B = exp(sqrt(0.2)*randn(1,N)-(2*0.2));
  % 2. Run the particle filter.
  for t=1:T
    % Compute the weights.
    w = exp((-0.5/(m.sig2/(m.r^2)))*((repmat(P(t),1,N) - ...
            (1/m.r).*(repmat(D(t),1,N)) - B).^2));
    w(w==0)=exp(-745);
    w = w/sum(w);
    % Save the particles and corresponding weights.
    xPWeighted(t,:) = B;
    wt(t,:) = w;
    % Compute state estimate, see Eq. (3.13).
    xf(t) = sum(w.*B);
    % Resample the particles, see sysresample.m
    % in Programming codes for Chapter 3.
    index = sysresample(w);
    B = B(index);
    xPResampled(t,:) = B;
    % Sequential importance sampling => new particles for t+1.
```

```
    idx1 = find(B <= m.tau);
    idx2 = find(B > m.tau);
    B(idx1) = (1+m.r)*B(idx1).*exp(sqrt(m.iota2)*...
              randn(1,length(idx1))-(0.5*m.iota2));
    B(idx2) = (m.kappa+((1+m.r)/((m.p)))*(B(idx2)-(m.kappa*(1/(1+...
              m.r)))).* binornd(1,(m.p),1,length(idx2))).*...
              exp(sqrt(m.iota2)*randn(1,length(idx2))-(0.5*m.iota2));
  end;
  pf.Xf = xf;
  pf.xPWeighted = xPWeighted;
  pf.xPResampled = xPResampled;
  pf.w = wt;
end
```

particle_smoother4.m

```
%%% Particle smoother
function ps = particle_smoother4(m,N,PF)
  T = size(PF.xPWeighted,1);
  wT = zeros(T,N);
  kk = ones(1,N);

   % 1. Set the smoothed weights in T.
  wT(T,:) = PF.w(T,:);
  % 2. Run the particle smoother.
  for t = T-1:-1:1
    p_xt1_xt= zeros(N,N);
    % Use the computed particles in t and t+1
    % form sequential importance sampling.
    xt = PF.xPWeighted(t,:);
    xt1 = PF.xPWeighted(t+1,:);
    idx1 = find(xt <= m.tau);
```

```
    idx2 = find(xt >  m.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
    % Compute the density of all possible particle combinations.
    p_xt1_xt(idx2,:) = (m.p)*lognpdf(repmat(xt1,length(idx2),1),...
                       (-(m.iota2/2) + log(m.kappa+((1+m.r)/(m.p))*...
                       (xti2-m.kappa*(1/(1+m.r))))),sqrt(m.iota2))+...
                       (1-(m.p))*lognpdf(repmat(xt1,length(idx2),1),...
                       (-(m.iota2/2) + log(m.kappa)),sqrt(m.iota2));
    p_xt1_xt(idx1,:) = lognpdf(repmat(xt1,length(idx1),1),...
                       (-(m.iota2/2)+log((1+m.r)*xti1)),sqrt(m.iota2));
   % Compute the (normalized) smoothed weights in Eq.(3.20).
   % 1. Compute the denominator of the summand in Eq. (3.20).
   vk = PF.w(t,:)*p_xt1_xt;
   % Set all weighted densities on minimal positive value to ensure
   % numerical stability in the next step.
   vk(vk==0) = exp(-745);
   % 2. Compute the complete sum in Eq. (3.20).
   wij(:,:,t) = (repmat(PF.w(t,:),N,1)'.*repmat(wT(t+1,:),N,1).*...
                 p_xt1_xt)./repmat(vk,N,1);
   k = (repmat(wT(t+1,:),N,1).*p_xt1_xt)./repmat(vk,N,1);
   sk = k*kk';
   % 3. Compute Eq.(3.20).
   wT(t,:) =  PF.w(t,:).*sk';
  end;
  ps.wT = wT;
  ps.wij = wij;
end
```

Q4_theta_k.m

```
function Qfun = Q4_theta_k(theta,mEst,xPWeighted,wij,wT,P,D)
  % Parameter for maximizing.
  r = theta(1);
  sig2 = theta(2);
  iota2 = theta(3);
  kappa = theta(4);
  p = theta(5);
  % Sample size.
  T = size(xPWeighted,1);
  % Number of particles.
  N = size(xPWeighted,2);
  Qfun3 = 0;
  Qfun2 = 0;
  kk = ones(1,N);
  % Compute the second term of the approximated Q-function
  % given in Eq.(3.30).
  for t=1:T-1
    p_xt1_xt_k= zeros(N,N);
    xt = xPWeighted(t,:);
    xt1 = xPWeighted(t+1,:);
    idx1 = find( xt <= mEst.tau);
    idx2 = find( xt > mEst.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
    p_xt1_xt_k(idx2,:) = (p)*lognpdf(repmat(xt1,length(idx2),1),...
                         (-(iota2^2/2)+log(kappa+((1+r)/(p))*...
                         (xti2-kappa*(1/(1+r))))),sqrt(iota2^2))+...
                         (1-(p))*lognpdf(repmat(xt1,length(idx2),1),...
                         (-(iota2^2/2) + log(kappa)),sqrt(iota2^2));
    p_xt1_xt_k(idx1,:) = lognpdf(repmat(xt1,length(idx1),1),...
                         (-(iota2^2/2)+log((1+r)*xti1)),sqrt(iota2^2));
```

```
    log_p_xt1_xt_k=log(p_xt1_xt_k);
    % To ensure numerical stability (20 times the computer tolerance).
    log_p_xt1_xt_k(log_p_xt1_xt_k == -Inf)= -15000;
    wijp = sum((wij(:,:,t).*(log_p_xt1_xt_k))*kk') ;
    Qfun2 = Qfun2 + wijp;
  end;
  % Compute the third term of the approximated Q-function given in
  % Eq.(3.30).
  for t=1:T
    p_yt_xt = -(1/2)*log((sig2/r)^2)-(1/(2*(sig2/r)^2))*(P(t)-...
              (1/r)*D(t)-xPWeighted(t,:)).^2 ;
    Qfun3 = Qfun3 + wT(t,:)*p_yt_xt';
  end;
  Qfun = Qfun3 + Qfun2 ;% Qfun1 is neglected;
  Qfun = -Qfun;
end
```

standard_error4.m

```
function SE = standard_error4(xPWeighted,wT,wij,P,D,m,N)
  T = size(xPWeighted,1);
  kk = ones(1,N);
  f11 = zeros(5,T);
  % Compute the second term of the score of the observed-data
  % log-likelihood function given in Eq. (3.40).
  for t = 2:T
    xt = xPWeighted(t-1,:);
    xt1 = xPWeighted(t,:);
    idx1 = find(xt <= m.tau);
    idx2 = find(xt > m.tau);
    xti1 = repmat(xt(idx1),N,1)';
    xti2 = repmat(xt(idx2),N,1)';
```

```
f1a = zeros(N,N);
f3a = zeros(N,N);
f4a = zeros(N,N);
f5a = zeros(N,N);
f1 = zeros(5,1);
% Compute the first derivatives of the joint log-likelihood
% functionwith respect to theta for all N paths. We compute
% these first derivatives with the mathematical software
% "Maple". First derivatives with respect to sig2 do not
% dependent on the latent bubble (particles).
% First derivative with respect to sig2.
f2a = 0.5*(-m.sig2+(P(t).^2-2.*P(t).*xt1+xt1.^2).*m.r.^2+(2.*...
      D(t).*xt1-2.*D(t).*P(t)).*m.r+D(t).^2)./(m.sig2.^2);
% Case 1: smoothed particle is smaller then tau.
% First derivative with respect to r.
f1a(idx1,:) = (log((repmat(xt1,length(idx1),1)))+(1./2).*m.iota2-...
              log((1+m.r).*xti1))./(m.iota2.*(1+m.r)) ;
% First derivative with respect to iota2.
f3a(idx1,:) = -(1./8).*(-4.*log((repmat(xt1,length(idx1),...
              1))).^2+8.*log((repmat(xt1,length(idx1),1))).*...
              log((1+m.r).*xti1)+m.iota2.^2-4.*log((1+m.r).*...
              xti1).^2+4.*m.iota2)./m.iota2.^2;
% First derivative with respect to kappa.
f4a(idx1,:) = 0;
% First derivative with respect to p.
f5a(idx1,:) = 0;
% Case 2: smoothed particle is bigger than tau.
% First derivative with respect to r: state density.
f1a(idx2,:) = ((1./2).*(2.*log((repmat(xt1,length(idx2),1)))+...
              m.iota2-2.*log((m.kappa.*m.p+xti2+xti2.*m.r-...
              m.kappa)./m.p)).*xti2.*exp(-(1./8).*(2.*...
              log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
              log((m.kappa.*m.p+xti2+xti2.*m.r-m.kappa)./...
```

```
                m.p)).^2./m.iota2).*m.p./(m.iota2.*(m.kappa.*m.p+...
                xti2+xti2.*m.r-m.kappa).*(m.p.*exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log((m.kappa.*m.p+xti2+xti2.*m.r-m.kappa)./...
                m.p)).^2./m.iota2)-exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(m.kappa)).^2./m.iota2).*m.p+exp(-(1./8).*...
                (2.*log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(m.kappa)).^2./m.iota2)));
% First derivative with respect to iota2.
f3a(idx2,:) = -(1./8).*(-m.p.*(-4.*...
                log((repmat(xt1,length(idx2),1))).^2+8.*...
                log((repmat(xt1,length(idx2),1))).*log(((-1+m.p).*...
                m.kappa+xti2.*(1+m.r))./m.p)+m.iota2.^2-4.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p).^2+...
                4.*m.iota2).*exp(-(1./8).*...
                (2.*log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)).^2./...
                m.iota2)+exp(-(1./8).*(2.*log((repmat(xt1,...
                length(idx2),1)))+m.iota2-2.*log(m.kappa)).^2./...
                m.iota2).*(8.*log((repmat(xt1,length(idx2),1))).*...
                log(m.kappa)+m.iota2.^2-4.*log(m.kappa).^2-4.*...
                log((repmat(xt1,length(idx2),1))).^2+4.*m.iota2).*...
                (-1+m.p))./(m.iota2.^2.*(-m.p.*exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)).^2./...
                m.iota2)+exp(-(1./8).*(2.*log((repmat(xt1,...
                length(idx2),1)))+m.iota2-2.*log(m.kappa)).^2./...
                m.iota2).*(-1+m.p)));
% First derivative with respect to kappa.
f4a(idx2,:) = (1./2).*(-1+m.p).*(-m.kappa.*m.p.*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)).*...
```

```
                exp(-(1./8).*(2.*log((repmat(xt1,length(idx2),1)))+...
                m.iota2-2.*log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./...
                m.p)).^2./m.iota2)+((-1+m.p).*m.kappa+xti2.*(1+...
                m.r)).*exp(-(1./8).*(2.*log((repmat(xt1,...
                length(idx2),1)))+m.iota2-2.*log(m.kappa)).^2./...
                m.iota2).*(2.*log((repmat(xt1,length(idx2),1)))+...
                m.iota2-2.*log(m.kappa))./(m.kappa.*((-1+m.p).*...
                m.kappa+xti2.*(1+m.r)).*(-m.p.*exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)).^2./...
                m.iota2)+exp(-(1./8).*(2.*log((repmat(xt1,...
                length(idx2),1)))+m.iota2-2.*log(m.kappa)).^2./...
                m.iota2).*(-1+m.p)).*m.iota2);
% First derivative with respect to p.
f5a(idx2,:) = (1./2).*((((-2-2.*m.r).*xti2+2.*m.kappa).*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)+((2+...
                2.*m.r).*xti2-2.*m.kappa).*log((repmat(xt1,...
                length(idx2),1)))-(2.*((1./2+(1./2).*m.r).*xti2+...
                m.kappa.*(m.p-1./2))).*m.iota2).*exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./m.p)).^2./...
                m.iota2)+(2.*((-1+m.p).*m.kappa+xti2.*(1+m.r))).*...
                exp(-(1./8).*(2.*log((repmat(xt1,length(idx2),1)))+...
                m.iota2-2.*log(m.kappa)).^2./m.iota2).*m.iota2)./...
                (((-1+m.p).*m.kappa+xti2.*(1+m.r)).*(-m.p.*exp(-...
                (1./8).*(2.*log((repmat(xt1,length(idx2),1)))+...
                m.iota2-2.*log(((-1+m.p).*m.kappa+xti2.*(1+m.r))./...
                m.p)).^2./m.iota2)+exp(-(1./8).*(2.*...
                log((repmat(xt1,length(idx2),1)))+m.iota2-2.*...
                log(m.kappa)).^2./m.iota2).*(-1+m.p)).*m.iota2);
% First derivative with respect to r: observation density.
f11a = (-(-m.sig2+(P(t).^2-2.*P(t).*xt1+xt1.^2).*m.r.^2+(-D(t)*...
       P(t)+D(t).*xt1).*m.r)./(m.sig2*m.r));
```

```
  % Compute the weighted sum of the derivatives (expectation value).
  f1(1) = sum((wij(:,:,t-1).*(f1a))*kk') + wT(t,:)*f11a';
  f1(2) = wT(t,:)*f2a';
  f1(3) = sum((wij(:,:,t-1).*(f3a))*kk') ;
  f1(4) = sum((wij(:,:,t-1).*(f4a))*kk') ;
  f1(5) = sum((wij(:,:,t-1).*(f5a))*kk') ;
  f1(isnan(f1))=0; % To ensure numerical stability.
  % Sum over t = 2, ..., T.
  f11(:,t)=f11(:,t)+(f1);
end
% First derivatives of the joint log-likelihood function at t=1.
xt = xPWeighted(1,:);
f1 = zeros(5,1);
% First derivative with respect to r.
f1a = -(-m.sig2+(P(1).^2-2.*P(1).*xt+xt.^2).*m.r.^2+(-D(1).*P(1)+...
      D(1).*xt).*m.r)./(m.sig2*m.r);
% First derivative with respect to sig2.
f2a = 0.5*(-m.sig2+(P(1).^2-2.*P(1).*xt+xt.^2).*m.r.^2+(2.*D(1).*...
      xt-2.*D(1).*P(1)).*m.r+D(1).^2)./(m.sig2.^2);
% First derivative with respect to iota2.
%f3a = 0;
% First derivative with respect to kappa.
%f4a = 0);
% First derivative with respect to p.
%f5a = 0;
% Compute the weighted sum of the first derivatives.
f1(1) = wT(1,:)*f1a';
f1(2) = wT(1,:)*f2a';
f1(3) = 0;
f1(4) = 0;
f1(5) = 0;
f11(:,1) = (f1);
% Expected value of the derivative of the joint log-likelihood.
```

```
  score = f11;
  % Take into account the dependence among the lagged terms by the
  % procedure of Newey and West (1987).
  A = 0;
  TT = size(score,2);
  % Compute the several covariance terms Aj given Eq. (3.42).
  for j = 0:(TT-1)
    A = 0;
    for t = 1:(TT-(j))
      A1 = score(:,t)*score(:,t+j)';
      A = A+A1;
    end
    Aj(:,:,(j+1)) = A;
  end
  format long
  % Compute the negative expected value of the Hessian, see Eq. (3.41).
  hessian = Aj(:,:,1);
  lag = 15;
  for j = 1:lag
    hessian = hessian+(1-j./(lag+1)).*(Aj(:,:,(j+1))+Aj(:,:,(j+1))');
  end
  % Compute the covariance matrix of the parameter estimates by taking
  % the inverse of the hessian.
  Cov = inv(hessian);
  % Standard errors are given by the diagonal of the covariance matrix.
  Cov_d = diag(Cov);
  % Results.
  SE.r = Cov_d(1);
  SE.sig2 = Cov_d(2);
  SE.iota2 = Cov_d(3);
  SE.kappa = Cov_d(4);
  SE.p = Cov_d(5);
end
```

Programming codes for Chapter 5

In the following we provide the MATLAB-Code for the identification and estimation of our economic nonlinear state-space model with periodic, stochastically deflating bubbles as introduced in Chapter 5. The main file main_chapter5.m gives the programming code for the simulation study of the economic model with our new bubble specification as well as the general programming code for the particle based EM algorithm to identify the nonlinear state-space model. After this, the file gives the general programming codes for evaluating the estimation results, that is the estimation of the latent bubble process, the computation of the standard errors and the goodness-of-fit test. At the end, the main file includes the code for the computation of the conditional stock-price volatility path.

At first we give the programming code of the main script. After that, the other m-files needed are given in the order of their occurrence in the main file. The codes for the EM algorithm via particle-filter methods are based on the descriptions in Schön et al. (2011).

main_chapter5.m

```
%%%%% Main script - Chapter 5
%%%%% Simulate data for the nonlinear state-space model

% Number of data.
T = 250;
% Parameter setting for the nonlinear state-space model.
% Set the required rate of return.
r = 0.02;
% Set the parameter of the observation equation.
m.phi = 50;
m.sig2 = 1.5;
% Set the parameter of the new periodic, stochastically deflating
% bubble mode (state equation).
```

```
m.psi = 1/(1+r);
m.iota2  = 0.02;
m.p = 0.87;
m.alpha = 0.91;

% Dividend process (driftless random walk with sd 0.03).
D = zeros(1,T); noise = randn(1,T);
% Set the starting value.
D(1)=0.3;
% Simulate data of the dividend process.
for t=1:(T-1)
  D(t+1) = D(t) + 0.03*noise(t);
end;

% Periodic, stochastically deflating bubble process.
Bt = zeros(1,T); y = randn(1,T); nu = binornd(1,m.p,1,T);
% Set the starting value.
Bt(1) = 0.5;
% Simulate data of the new bubble.
for t=1:T-1
  Bt(t+1) = (((m.alpha/(m.psi*m.p)-(1-m.alpha)/(m.psi*(1-m.p)))*...
             nu(t)+(1-m.alpha)/(m.psi*(1-m.p)))*Bt(t)).*...
             exp(sqrt(m.iota2)*y(t)-(0.5*m.iota2));
end;

% Calculate the stock-price process.
varepsilon = randn(1,T);
P = m.phi*D + Bt + sqrt(m.sig2)*varepsilon;

% Save the data in the structure array z.
z.P     = P;
z.D     = D;
z.BTrue = Bt(1:T);
```

```
%%%%% Parameter estimation
%%%%% EM algorithm using the particle filter and particle smoother
% We need the data set given in the structure array z. For the
% artificial and real-world data sets, read the (real) prices and
% (real) dividends from the corresponding Excel file and save the
% prices as z.P and dividends as z.D in the structure array z.

% Set max. number of iterations, if no convergence criterion is used!
% opt.maximumiter = 500;
% Set the number of particles for the EM algorithm.
N  = 300;
% Set the parameter vector theta_0 to initiate the EM algorithm.
phi0 = 30;
sig20 = 0.5 ;
psi0 = 0.85;
iota20 = 0.01;
p0 = 0.7;
alpha0 = 0.75;
theta0 = [phi0 sqrt(sig20) psi0 sqrt(iota20) p0 alpha0];
% Define the vector for the parameter estimates.
mEst = m;
mEst.phi = theta0(1);
mEst.sig2 = theta0(2)^2;
mEst.psi = theta0(3);
mEst.iota2 = theta0(4)^2;
mEst.p = theta0(5);
mEst.alpha = theta0(6);

% Store the iterates.
theta = theta0;
theta_k(1,1:6) = theta0;
mEst_k(1) = mEst;
crit_value(1) = 1;
```

```
% Set constraints on the parameter vector.
% The lower bound of alpha and p is set to 0.5 to define the
% 1 phase as the state of growth and state 2 as the
% incomplete bursting state.
lb = [0.000001,0.000001,0.000001,0.000001,0.5,0.5];
ub = [250,8,0.999999,5,0.999999,1];
% Define some optimization conditions.
options = optimset('Algorithm','active-set','Display','off',...
                   'TolFun',1e-5,'TolX',1e-5);
% Fix the random generator to start each iteration with the same
% random numbers. This helps to minimize some random effects of the
% Monte Carlo approach.
stream = RandStream.getGlobalStream;
% General Setting: savedState = stream.State;
% To reproduce the results use this random number stream.
savedState = uint32(xlsread('savedState'));
% Use loop if no termination condition is used!
% for k = 1:opt.maximumiter
% Or define the convergence criterion.
c = 1/(N);

%%% Start of the EM algorithm.
k = 2;
while (crit_value(k-1) > c)
  % Set the random stream in each iteration k.
  stream.State = savedState;
  % Run the particle filter and particle smoother to obtain
  % particles from the sequential importance sampling and
  % the corresponding smoothed weights.
  %%% Expectation step at iteration k.
  % Run the particle filter.
  PF = particle_filter5(mEst,N,z);
  % Run the particle smoother.
```

```
  PS = particle_smoother5(mEst,N,PF);
  % Calculate the Q-function subject to theta_k.
  Q_value_k_k = -(Q5_theta_k(theta,PF.xPWeighted,PS.wij,PS.wT,z.P,z.D));
  %%% Maximization step at iteration k.
  % Maximize the Q-function subject to theta and obtain theta_k+1.
  [theta,Qval] = fmincon('Q5_theta_k',theta,[],[],[],[],lb,ub,[],...
                    options,PF.xPWeighted,PS.wij,PS.wT,z.P,z.D);
  % Calculate the Q-function subject to theta_k+1.
  Q_value_k_k1 = -Qval;
  % Store the parameter estimates in iteration k.
  mEst.phi = theta(1);
  mEst.sig2 = theta(2)^2;
  mEst.psi = theta(3);
  mEst.iota2 = theta(4)^2;
  mEst.p = theta(5);
  mEst.alpha = theta(6);
  % Store the iterates.
  theta_k(k,1:6)= theta;
  mEst_k(k) = mEst;
  % Compute critical value to check the convergence criterion.
  crit_value(k) = Q_value_k_k1-Q_value_k_k;
  % Print the parameter estimates.
  disp(['Iteration nr: ' num2str((k-1))...
        ' Estimates, phi: ' num2str(mEst.phi),...
        ' sig2: ',num2str(mEst.sig2),'  psi: ',num2str(mEst.psi),...
        ' iota2: ',num2str(mEst.iota2),' p: ',num2str(mEst.p),...
        ' alpha: ',num2str(mEst.alpha)])
  % If crit_value(k) > c: k is updated to k+1 and a new iteration
  % is conducted.
  k = k+1;
end
```

```
%%%%% Evaluating the estimation results
% Fix the random generator to reproduce the results.
stream = RandStream.getGlobalStream;
savedState = uint32(xlsread('savedState'));
stream.State = savedState;

%%% Estimation of the latent Evans-bubble Process
% Set number of particles.
N=500;
% Use the estimated parameter set.
mEst = mEst_k(k-1);
% Compute the particles and smoothed weights.
% Run the particle filter.
PF = particle_filter5(mEst,N,z);
% Run the particle smoother.
PS = particle_smoother5(mEst,N,PF);
% Compute the smoothed states.
for t=1:T
  xs(t) = sum(PS.wT(t,:).*PF.xPWeighted(t,:));
end;

%%% Standard errors
% Compute the standard errors of the parameter estimates by the use
% of the stable estimator of the information matrix established by
% Duan and Fulop (2011).
standard_error5(PF.xPWeighted,PS.wT,PS.wij,z.P,z.D,mEst,N)

%%% Goodness-of-fit test
% Compute the PITs and the corresponding empirical cdf,
% see forecast_denstiy.m in Programming codes for Chapter 3.
pit = forecast_density(PF,mEst,z);
[cdf_pit]=ecdf(pit);
```

```
% KS-test.
% Simulate 1 million uniforms.
r_unif = unifrnd(0,1,1000000,1);
% Compare these uniforms with the PITs by using a KS-Test.
[h,pValue,ks_stat] = kstest2(r_unif,pit)

% LB-test.
[h,pValue,lb_stat,cValue] = lbqtest(pit)

%%% Conditional stock-price variance with the periodic,
%%% stochastically deflating bubble
% Compute the conditional variance given in Eq. (5.15).
Var_t = zeros(1,T);
beta=1/r;
% We assume that dividends follow a driftless random walk with
% variance = 0.0009.
var_div = 0.0009;
% As bubble component we use the simulated process of the new
% bubble displayed in Figure 5.3 with the parameters psi = 0.9804,
% iota2 = 0.02, p = 0.87 and alpha = 0.91.
Bt = xlsread('b6n')';
for t=2:T
  Var_t(t) = (m.alpha-m.p)^2/(m.psi*m.p*((1-m.p)^2))*...
             (exp(m.iota2)-m.p)*(Bt(t-1)^2)+ (((1-m.alpha)^2+2*...
             (m.alpha-m.p)*(1-m.alpha))/((m.psi^2)*(1-m.p)^2))*...
             (exp(m.iota2)-1)*(Bt(t-1)^2)+(beta^2*var_div);
end;
```

particle_filter5.m

```
%%% Particle filter
function pf = particle_filter5(m,N,z)
  P = z.P;
  D = z.D;
  T = size(P,2);
  xf = zeros(1,size(P,2));
  xPWeighted = zeros(T,N);
  xPResampled = zeros(T,N);
  wt = zeros(T,N);

  % 1. Initialize particles.
  % Initial state unknown, assumed to be a positive random variable.
    B = exp(sqrt(0.2)*randn(1,N)-(2*0.2));
  % 2. Run the particle filter.
  for t=1:T
    % Compute the weights.
    w = exp((-0.5/m.sig2)*((repmat(P(t),1,N) -...
            m.phi.*(repmat(D(t),1,N)) - B).^2));
    w(w==0) = exp(-745);
    w = w/sum(w);
    % Save the particles and corresponding weights.
    xPWeighted(t,:) = B;
    wt(t,:) = w;
    % Compute state estimate, see Eq. (3.13).
    xf(t) = sum(w.*B);
    % Resample the particles, see sysresample.m in Programming
    % codes for Chapter 3.
    index = sysresample(w);
    B = B(index);
    xPResampled(t,:) = B;
    % Sequential importance sampling => new particles for t+1.
```

```
    B =  (((m.alpha/(m.psi*m.p)-(1-m.alpha)/(m.psi*(1-m.p))).*...
         binornd(1,(m.p),1,N)+(1-m.alpha)/(m.psi*(1-m.p))).*B).*...
         exp(sqrt(m.iota2)*randn(1,N)-(0.5*m.iota2));
  end;
  pf.Xf = xf;
  pf.xPWeighted = xPWeighted;
  pf.xPResampled = xPResampled;
  pf.w = wt;
end
```

particle_smoother5.m

```
%%% Particle smoother
function ps = particle_smoother5(m,N,PF)
  T = size(PF.xPWeighted,1);
  wT = zeros(T,N);
  kk = ones(1,N);

  % 1. Set the smoothed weights in T.
  wT(T,:) = PF.w(T,:);
  % 2. Run the particle smoother.
  for t = T-1:-1:1
    p_xt1_xt = zeros(N,N);
    % Use the computed particles in t and t+1
    xt = PF.xPWeighted(t,:);
    xt1 = PF.xPWeighted(t+1,:);
    xti = repmat(xt,N,1)';
    % Compute the density of all possible particle combinations.
    p_xt1_xt = (m.p)*lognpdf(repmat(xt1,N,1),(-(m.iota2/2)+...
              log((m.alpha/(m.p*m.psi))*(xti))),sqrt(m.iota2))+...
              (1-(m.p))*lognpdf(repmat(xt1,N,1),(-(m.iota2/2)+...
              log(((1-m.alpha)/((1-m.p)*m.psi))*(xti))),sqrt(m.iota2));
```

```
   % Compute the (normalized) smoothed weights in Eq.(3.20).
   % 1. Compute the denominator of the summand in Eq. (3.20).
   vk = PF.w(t,:)*p_xt1_xt;
   % Set all weighted densities which are approx. zero on minimal
   % positive value to ensure numerical stability in the next step.
   vk(vk==0) = exp(-745);
   % 2. Compute the complete sum in Eq. (3.20)
   wij(:,:,t) = (repmat(PF.w(t,:),N,1)'.*repmat(wT(t+1,:),N,1).*...
                 p_xt1_xt)./repmat(vk,N,1);
   k = (repmat(wT(t+1,:),N,1).*p_xt1_xt)./repmat(vk,N,1);
   sk = k*kk';
   % 3. Compute Eq.(3.20)
   wT(t,:) =  PF.w(t,:).*sk';
  end;
  ps.wT = wT;
  ps.wij = wij;
end
```

Q5_theta_k.m

```
function Qfun = Q5_theta_k(theta,xPWeighted,wij,wT,P,D)
  % Parameter for maximizing.
  phi = theta(1);
  sig2 = theta(2);
  psi = theta(3);
  iota2 = theta(4);
  p = theta(5);
  alpha = theta(6);
  % Sample size.
  T = size(xPWeighted,1);
  % Number of particles.
  N = size(xPWeighted,2);
```

```
  Qfun3 = 0;
  Qfun2 = 0;
  kk = ones(1,N);

  % Compute the second term of the approximated Q-function given in
  % Eq.(3.30).
  for t=1:T-1
    p_xt1_xt_k = zeros(N,N);
    xt = xPWeighted(t,:);
    xt1 = xPWeighted(t+1,:);
    xti = repmat(xt,N,1)';
     p_xt1_xt_k = (p)*lognpdf(repmat(xt1,N,1),(-(iota2^2/2)+...
                 log(((alpha)/((p)*(psi)))*(xti))),sqrt(iota2^2))+...
                 (1-(p))*lognpdf(repmat(xt1,N,1),(-(iota2^2/2)+...
                 log(((1-(alpha))/((1-(p))*((psi))))*...
                 (xti))),sqrt(iota2^2));
    log_p_xt1_xt_k=log(p_xt1_xt_k);
    % To ensure numerical stability (20 times the computer tolerance).
    log_p_xt1_xt_k(log_p_xt1_xt_k == -Inf) = -15000;
    wijp = sum((wij(:,:,t).*(log_p_xt1_xt_k))*kk') ;
    Qfun2 = Qfun2 + wijp;
  end;
  % Compute the third term of the approximated Q-function given in
  % Eq.(3.30).
  for t=1:T
    p_yt_xt = -(1/2)*log(sig2^2)-(1/(2*sig2^2))*(P(t)-phi*D(t)-...
              xPWeighted(t,:)).^2 ;
    Qfun3 = Qfun3 + wT(t,:)*p_yt_xt';
  end;
  Qfun = Qfun3 + Qfun2 ;% Qfun1 is neglected.
  Qfun = -Qfun;
end
```

standard_error5.m

```
function SE = standard_error5(xPWeighted,wT,wij,P,D,m,N)
  T = size(xPWeighted,1);
  kk = ones(1,N);
  f11 = zeros(6,T);

  % Compute the second term of the score of the observed-data
  % log-likelihood function given in Eq. (3.40).
  for t = 2:T
    xt = xPWeighted(t-1,:);
    xt1 = xPWeighted(t,:);
    xti = repmat(xt,N,1)';
    xtj = repmat(xt1,N,1);
    f1 = zeros(6,1);
    % Compute the first derivatives of the joint log-likelihood
    % function with respect to theta for all N paths. We compute
    % these derivatives with the mathematical software "Maple".
    % First derivative with respect to phi.
    f1a = (P(t)-m.phi.*D(t)-xt1).*D(t)./m.sig2 ;
    % First derivative with respect to sig2.
    f2a = -0.5.*(-m.phi.^2.*D(t).^2+2.*D(t).*((-xt1)+P(t)).*...
         m.phi-P(t).^2-(xt1).^2+2.*P(t).*(xt1)+m.sig2)./...
         (sqrt(m.sig2).*sqrt(m.sig2.^3));
    % First derivative with respect to psi.
    f3a = 0.5.*((-1+m.p).*(2.*log(xtj)+m.iota2-2.*log((-1+m.alpha).*...
         xti./(m.psi.*(-1+m.p)))).*exp(-(1./8).*(2.*log(xtj)+...
         m.iota2-2.*log((-1+m.alpha).*xti./(m.psi.*(-1+m.p)))).^2./...
         m.iota2)-exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log(m.alpha.*...
         xti./(m.psi.*m.p))).^2./m.iota2).*m.p.*(2.*log(xtj)+...
         m.iota2-2.*log(m.alpha.*xti./(m.psi.*m.p)))./...
         (((1-m.p).*exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log((-1+...
         m.alpha).*xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+m.p.*...
```

```
     exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./...
     (m.psi.*m.p))).^2./m.iota2)).*m.psi.*m.iota2);
% First derivative with respect to iota2.
f4a = 0.5.*( -(-1+m.p).*(log(xtj).^2-2.*log(xtj).*log((-1+...
     m.alpha).*xti./(m.psi.*(-1+m.p)))-(1./4).*m.iota2.^2+...
     log((-1+m.alpha).*xti./(m.psi.*(-1+m.p))).^2-m.iota2).*...
     exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log((-1+m.alpha).*...
     xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+m.p.*exp(-(1./8).*...
     (2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./(m.psi.*...
     m.p))).^2./m.iota2).*(log(xtj).^2-2.*log(xtj).*...
     log(m.alpha.*xti./(m.psi.*m.p))-(1./4).*m.iota2.^2+...
     log(m.alpha.*xti./(m.psi.*m.p)).^2-m.iota2))./(m.iota2.^2.*...
     ((1-m.p).*exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log((-1+...
     m.alpha).*xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+m.p.*...
     exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./...
     (m.psi.*m.p))).^2./m.iota2)));
% First derivative with respect to p.
f5a = 0.5.*((-m.iota2+2.*log(xtj)-2.*log((-1+m.alpha).*xti./...
     (m.psi.*(-1+m.p)))).*exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*...
     log((-1+m.alpha).*xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+2.*...
     exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./...
     (m.psi.*m.p))).^2./m.iota2).*((1./2).*m.iota2-log(xtj)+...
     log(m.alpha.*xti./(m.psi.*m.p)))./(m.iota2.*((1-m.p).*...
     exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log((-1+m.alpha).*...
     xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+m.p.*exp(-(1./8).*...
     (2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./...
     (m.psi.*m.p))).^2./m.iota2)));
 % First derivative with respect to alpha.
 f6a = 0.5.*((2.*(-log(xtj)-(1./2).*m.iota2+log((-1+m.alpha).*...
      xti./(m.psi.*(-1+m.p))))).*(-1+m.p).*m.alpha.*exp(-(1./8).*...
      (2.*log(xtj)+m.iota2-2.*log((-1+m.alpha).*xti./(m.psi.*...
      (-1+m.p)))).^2./m.iota2)-2.*exp(-(1./8).*(2.*log(xtj)+...
      m.iota2-2.*log(m.alpha.*xti./(m.psi.*m.p))).^2./m.iota2).*...
```

```
        m.p.*(-1+m.alpha).*(-log(xtj)-(1./2).*m.iota2+log(m.alpha.*...
        xti./(m.psi.*m.p)))./(m.iota2.*(-1+m.alpha).*m.alpha.*...
        ((1-m.p).*exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log((-1+...
        m.alpha).*xti./(m.psi.*(-1+m.p)))).^2./m.iota2)+m.p.*...
        exp(-(1./8).*(2.*log(xtj)+m.iota2-2.*log(m.alpha.*xti./...
        (m.psi.*m.p))).^2./m.iota2)));
  % Compute the weighted sum of the derivatives (expected value).
  f1(1) = wT(t,:)*f1a';
  f1(2) = wT(t,:)*f2a';
  f1(3) = sum((wij(:,:,t-1).*(f3a))*kk') ;
  f1(4) = sum((wij(:,:,t-1).*(f4a))*kk') ;
  f1(5) = sum((wij(:,:,t-1).*(f5a))*kk') ;
  f1(6) = sum((wij(:,:,t-1).*(f6a))*kk') ;
  f1(isnan(f1))=0; % To ensure numerical stability.
  % Sum over t = 2, ..., T.
  f11(:,t)=f11(:,t)+(f1);
end

% First derivatives of the joint log-likelihood function at t=1.
xt = xPWeighted(1,:);
f1 = zeros(6,1);
% First derivative with respect to phi.
f1a = (P(1)-m.phi.*D(1)-xt).*D(1)./m.sig2 ;
% First derivative with respect to sig2.
f2a = -0.5.*(-m.phi.^2.*D(1).^2+2.*D(1).*(-xt+P(1)).*m.phi-...
      P(1).^2-xt.^2+2.*P(1).*xt+m.sig2)./(sqrt(m.sig2).*...
      sqrt(m.sig2.^3));
% First derivative with respect to psi.
%f3a = 0;
% First derivative with respect to iota2.
%f4a = 0);
% First derivative with respect to kappa.
%f5a = 0;
```

```
% First derivative with respect to p.
%f6a = 0;
% Compute the weighted sum of the first derivatives.
f1(1) = wT(1,:)*f1a';
f1(2) = wT(1,:)*f2a';
f1(3) = 0;
f1(4) = 0;
f1(5) = 0;
f1(6) = 0;
f11(:,1) = (f1);
% Expected value of the derivative of the joint log-likelihood.
score = f11;
% Take into account the dependence among the lagged terms by the
% procedure of Newey and West (1987).
A = 0;
TT = size(score,2);
% Compute the several covariance terms Aj given Eq. (3.42).
for j = 0:(TT-1)
  A = 0;
  for t = 1:(TT-(j))
    A1 = score(:,t)*score(:,t+j)';
    A = A+A1;
  end
  Aj(:,:,(j+1)) = A;
end
format long
% Compute the negative expected value of the Hessian, see Eq. (3.41).
hessian = Aj(:,:,1);
lag = 15;
for j = 1:lag
  hessian = hessian+(1-j./(lag+1)).*(Aj(:,:,(j+1))+Aj(:,:,(j+1))');
end
```

```
  % Compute the covariance matrix of the parameter estimates by taking
  % the inverse of the hessian.
  Cov = inv(hessian);
  % Standard errors are given by the diagonal of the covariance matrix.
  Cov_d = diag(Cov);
  % Results.
  SE.phi = Cov_d(1);
  SE.sig2 = Cov_d(2);
  SE.psi = Cov_d(3);
  SE.iota2 = Cov_d(4);
  SE.p = Cov_d(5);
  SE.alpha = Cov_d(6);
end
```